Community Detection in Multi-Layer Networks Using Joint Nonnegative Matrix Factorization

Xiaoke Ma, Di Dong, and Quan Wang

Abstract—Many complex systems are composed of coupled networks through different layers, where each layer represents one of many possible types of interactions. A fundamental question is how to extract communities in multi-layer networks. The current algorithms either collapses multi-layer networks into a single-layer network or extends the algorithms for single-layer networks by using consensus clustering. However, these approaches have been criticized for ignoring the connection among various layers, thereby resulting in low accuracy. To attack this problem, a quantitative function (multi-layer modularity density) is proposed for community detection in multi-layer networks. Afterward, we prove that the trace optimization of multi-layer modularity density is equivalent to the objective functions of algorithms, such as kernel K-means, nonnegative matrix factorization (NMF), spectral clustering and multi-view clustering, for multi-layer networks, which serves as the theoretical foundation for designing algorithms for community detection. Furthermore, a Semi-Supervised joint Nonnegative Matrix Factorization algorithm (*S2-jNMF*) is developed by simultaneously factorizing matrices that are associated with multi-layer networks. Unlike the traditional semi-supervised algorithms, the partial supervision is integrated into the objective of the S2-jNMF algorithm. Finally, through extensive experiments on both artificial and real world networks, we demonstrate that the proposed method outperforms the state-of-the-art approaches for community detection in multi-layer networks.

Index Terms—Multi-layer networks, community structure, nonnegative matrix factorization, semi-supervised clustering

1 Introduction

Networks have emerged as effective tools for modeling and analyzing complex systems. Various networks exist in the real world, such as social networks [1], technological networks [2], and biological networks [3]. Network analysis has been extensively studied to reveal critical pattern that facilitates the understanding of networks. For example, the engagement of users in social networks can be used to model their behavior of [4], [5] and to predict whether or when a topic is prevalent [6], [7].

As one of the most important patterns in networks, a community (module) refers to a subgraph with more edges connecting vertices of the same group and fewer edges joining vertices of different groups. Complex networks have an ubiquitous community structure. For example, communities in protein interaction networks are the protein complexes that regulate gene expression [8]. Thus, it is desirable to detect communities in networks because they provide insights into the structure and function of the overall system. To quantify the community in networks, Newman et al. [9] proposed the well-known modularity Q, which implicitly defines communities based on the observation that random networks are not expected to exhibit a modular structure. Many algorithms have been developed by optimizing modularity Q or modularity density Q_D [10], [11], [12], [13], [14], [15], [16], [17]. More literatures can be referred to Ref. [18].

However, these algorithms are deliberately designed for single-layer networks where vertices are placed adjacent to each other by a single type of edges that encapsulate all connections among these vertices. This assumption is almost always a gross oversimplification, which can lead to misleading results and may even the fundamental inability to address certain problems. Recent advances in the study of networked systems have shown that the interconnected world is composed of networks that are coupled to one another through different layers, where each layer represents one of many possible subsystems or types of interactions [19], [20], [21], [22], [23]. For example, individuals have multiple of relationships in social networks, such as economic, political, and financial. In biology, protein interaction networks consist of seven distinct layers that account for different genetic and physical interactions with each layer containing thousands of protein-protein interactions [21].

Furthermore, analyzing multi-layer networks is of great importance because many interesting patterns that cannot be obtained by analyzing single-layer networks [24], [25], [26], [27]. For example, in social networks [28] such as Facebook, each vertex represents a user and each edge corresponds to a link between users. Usually, all links between users are assumed to take place at the same level (layer). However, the friendships in social networks may result from relationships of origins, such as colleagues in occupations, members of

- *X. Ma and Q. Wang are with the School of Computer Science and Technology, Xidian University, Xi'an, Shaanxi 710071, China. E-mail: {xkma, qwang}@xidian.edu.cn.*
- *D. Dong is with the Key Laboratory of Molecular Imaging, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, P.R. China. E-mail: di.dong@ia.ac.cn.*

Manuscript received 23 Jan. 2017; revised 21 Mar. 2018; accepted 28 Apr. 2018. Date of publication 1 May 2018; date of current version 9 Jan. 2019. (Corresponding author: Xiaoke Ma.)
Recommended for acceptance by X. Lin.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TKDE.2018.2832205

clubs. Given the problem of spreading information across networks, users initially select layers that they believe to contain information, and then begin spreading such information across these layers. As a schematic example, the protein-protein interaction networks in a biological system comprise vertices that denote proteins and edges that denote interactions. The interactions among proteins result from different clinical cancer stages [29]. When diagnosing cancers, the interactions at the early cancer stages must be selected. When conducting cancer therapy while for the therapy of cancers, the interactions from the middle or late cancer stages must be selected. Other examples of multi-layer networks in natural systems, such as transportation networks, can be found in Ref. [24].

Communities in single-layer networks comprise a group of well-connected vertices, while communities in multi-layer networks comprise a group of well-connected vertices in all layers. Unlike single-layer communities, multi-layer communities reveal the relations among vertices in various layers. For example, in multi-layer cancer networks where each layer corresponds to a specific clinical stage, a community represents a biological pathway that is critical for cancer diagnosis and therapy [29]. Moreover, in multi-layer collaboration networks, community corresponds to a group of scientists with stable collaborative relation [30]. Thus, detecting community in multi-layer networks is necessary. However, it is highly non-trivial to extract communities in multi-layer networks because of two reasons. First, multi-layer communities cannot be easily quantified because analyses on these multi-layer networks remain lacking. Second, complexity of multiple networks poses a great challenge on designing algorithms for identifying communities in multi-layer networks.

Despite these challenges, great efforts have been devoted to the extraction of multi-layer communities. The most intuitive way to discover multi-layer communities is to extend the algorithms for single-layer networks, which can be categorized into two classes, namely single-layer-network-analysis-based approaches [31], [32] and multi-layer-networks-analysis-based methods [33], [34], [35], [36], [37], [38], [39], [40]. Although very few attempts have been devoted to multi-layer community detection, many algorithms have been developed to discover single-layer communities. Thus, it is natural to detect multi-layer communities by extending the algorithms for single-layer community detection. Two strategies can be employed for the extension. The first strategy collapses the multi-layer networks into a single-layer network and then applies single-layer network algorithms to obtain the communities in the collapsed network [31]. Meanwhile the second strategy applies single-layer network algorithms to obtain the communities for each layer, and then combines the communities at various layers by using consensus clustering [32]. However, these algorithms are criticized for their low accuracy because they either cannot preserve the community structure in compressed networks or ignore the connection among various layers.

To overcome these problems, we must simultaneously take into account multiple layers. Ma et al. [29] developed the *M-Module* algorithm to extract communities in multi-layer networks based on a greedy search. Didier et al. [40] quantified the community in multi-layer networks by extending the modularity Q, and then developed the *MolTi* method for multi-layer communities based on topological of networks. By considering each layer of networks as a view, the multi-view clustering algorithms focus on how to combine various views to improve clustering performance. [33], [34], [36], [37], [38], [39]. For instance, Kumar et al. [38] developed the multi-view spectral clustering algorithm (*MVspec*), where a clustering solution is derived from each individual view and then all the solutions are fused together based on consensus. Han et al. [39] proposed the NMF-based multi-view algorithm (*MVnmf*), which formulates a joint matrix factorization process to extract the communities in multi-layer networks.

Although efforts have been devoted to community detection in multi-layer networks, many problems remain unsolved, such as how to characterize and quantify communities in multi-layer networks and what is the relation of algorithms for community detection in multi-layer networks. The current algorithms for community detection in multi-layer networks make use of either topological information or low-rank features. Thus, can we combine both topological and low-rank features to further improve the performance of algorithms for community detection in multi-layer networks? In this study, we address the first problem by presenting a quantitative function for community in multi-layer networks (called multi-layer modularity density). Afterward, we prove the equivalence of the objective functions of kernel K-means, NMF, spectral clustering, multi-view clustering and multi-layer modularity density. We also develop a novel semi-supervised joint nonnegative matrix factorization (S2-jNMF) algorithm by incorporating the priori information into NMF, which improves the accuracy of algorithms without increasing the time complexity.

The contributions of our work are summarized as follows:

- We propose a quantification function for community in multi-layer networks. This function is suitable for both weighted and unweighted multi-layer networks.
- We prove the equivalence among the objective function of kernel K-means, NMF, spectral clustering, multi-view clustering, and multi-layer modularity density, which provide a theoretical foundation for the multi-layer community detection algorithms.
- We propose the S2-jNMF algorithm by integrating partial supervision and NMF. This algorithm serves as a general framework for semi-supervised clustering for community detection in multi-layer networks. This algorithm also improves accuracy dramatically without increasing time complexity. The experiments demonstrate that the S2-jNMF algorithm outperforms the state-of-the-art methods.

The paper is organized as follows. Section 2 briefly reviews the related works. The multi-layer modularity density is proposed in Section 3. The equivalence relation is proved in Section 4. Section 5 is devoted to the S2-jNMF algorithm. The experiments and conclusion are presented in Sections 6 and 7, respectively.

2 Related Work

The traditional single-layer network assumes that all the links (edges) between vertices take place at the same level, which is too big a constraint that may occasionally result in the failure to fully capture the details of complex systems. For example, in web image retrieval, the visual information of images needs to be separated from their textual tags. Therefore, the multi-layer networks have received an increasing amount of attention because they can precisely characterize and model systems in real world [24].

Community detection in networks has become a hot topic since communities shed light on structure-function relations, which has been extensively studied in the single-layer networks. The most straightforward strategy is to extend the single-layer community detection algorithms to the multi-layer networks to develop network-compression- and consensus-based approaches. Network-compression-based approaches compress multi-layer networks into a single-layer network in which the single-layer community detection algorithms are adopted [31]. However, these methods cannot preserve the community structure in multi-layer networks, thereby leading to the low accuracy of algorithms [25]. Thus, it is promising to extract community without collapsing multi-layer networks. In this case, the single-layer community detection algorithms are independently applied to each layer, and then the communities at various layers are combined to establish a consensus community structure for multi-layer networks. However, this strategy is also criticized for ignoring the connection among various layers.

Thus, there is a critical need to develop effective algorithms for community detection in multi-layer networks, rather than by simply extending the available single-layer network algorithms. To identify communities in multi-layer networks, we must simultaneously take into account multiple layers during the community search procedure. The first step in multi-layer community detection is to quantify the community in multi-layer networks. Ma et al. [29] quantified the connectivity of communities in multi-layer networks using information entropy and transformed the community detection in multi-layer networks into an entropy optimization problem. They also proposed a greedy-search-based algorithm (*M-Module*) for multi-layer community detection. Didier et al. [40] quantified communities by extending the modularity function to detect communities from multi-layer networks and developed the (*MolTi*) algorithm for multi-layer communities. Some alternatives based on random network models are also available for the quantification of multi-layer communities [41], [42]. The successful application of these algorithms highlights the need for the integrative analysis of multiple layers, which is also one of the major motivations of this study.

In other words, the layers of networks can be considered as different views of data, which provide information complementary to each other. To integrate information from multiple views, multi-view clustering algorithms cluster multiple views simultaneously to derive a solution that uncovers the common latent structure shared by multiple views. Many multi-view clustering algorithms has been widely developed [33], [34], [35], [36], [37], [38], [39]. Existing multi-view clustering methods can be roughly categorized into two classes: loss-function and subspace-based approaches. The algorithms in the first category incorporate multiple views into the clustering process by optimizing the predefined loss functions [33], [38], while the subspace-based algorithms project multiple views into a common lower dimensional subspace where communities are discovered [35]. For example, to explore the information in each view, Kumar et al. [38] developed the multi-view spectral clustering *MVspec* algorithm, where a clustering solution is derived from each individual view and then all the solutions are fused based on consensus. However, MVspec is criticized for the independence of features from various views. To solve this problem, Han et al. [39] proposed *MVnmf* by formulating a joint matrix factorization process with the constraint that pushes the clustering solution of each view toward a common consensus instead of fixing such solution directly.

TABLE 1
Main Symbols

Symbol	Definition and Description
G	graph with vertex set V and edge set E
$\mathcal{G}$	multi-layer network $\{G^{[1]},\ldots,G^{[m]}\}$
$W^{[l]}$	the weighted adjacency matrix for $G^{[l]}$
$D^{[l]}$	degree diagonal matrix $D^{[l]} = diag(d_1^{[l]},\ldots,d_n^{[l]})$
$\mathcal{W}$	adjacency matrix $= \{W^{[1]},\ldots,W^{[m]}\}$
w_{ij}	the element at ith row jth column in matrix W
$\mathbf{w}_{i.}$	the ith row of matrix W
$\mathbf{w}_{.j}$	the jth column of matrix W
X	the indication matrix for V, where x_{ij}=1 if v_i belongs to cluster C_j, 0 otherwise
$X^{'}$	the transpose of matrix X
$\tilde{X}$	column normalized matrix X, i.e., $\mathbf{x}_{.j}/\sqrt{\mathbf{x}^{'}_{.j}\mathbf{x}_{.j}}$
l	the index for the layers of network $l \in \{1,\ldots,m\}$
$\{C_c\}_{c=1}^{k}$	the communities for $\mathcal{G}$
$Q_D^{\mathcal{G}}(\{C_i\}_{i=1}^{k})$	multi-layer modularity density for communities $\{C_i\}_{i=1}^{k})$ in $\mathcal{G}$ ($Q_D^{\mathcal{G}}$ for short)
$trace(W)$	the trace of matrix W, i.e., $trace(W) = \sum_i w_{ii}$

Even though great efforts have been devoted to community detection in multi-layer networks, few attempt has been made to draw the relation among various algorithms. In the forthcoming sections, we address the equivalence relation among algorithms.

3 Multi-Layer Modularity Density

Before giving a detailed description of multi-layer modularity density, we introduce some terminologies that are widely used in the forthcoming sections. The notations to define the model and algorithm are summarized in Table 1.

Given an undirected and unweighted network $G = (V, E)$ with vertex set $V = \{v_1, v_2, \ldots, v_n\}$ (n is the number of vertices in G, i.e., $n = |V|$) and edge set $E = \{(v_i, v_j)\}$, the adjacency matrix $A = (a_{ij})$ is constructed whose element is $a_{ij} = 1$ if there is an edge between vertex v_i and v_j, 0 otherwise. The degree of vertex v_i is defined as the number of neighbors in the network, i.e., $d_i = \sum_j a_{ij}$. If G is weighted, the weighted adjacency matrix is denoted by $W = (w_{ij})$, where element w_{ij} is the weight on edge (v_i, v_j). The weighted degree of vertex v_i is the sum of weight on edge connecting to v_i, i.e., $d_i = \sum_j w_{ij}$. For a pair of subsets V_1 and V_2 of V, let $L(V_1, V_2) = \sum_{i\in V_1, j\in V_2} w_{ij}$ and $\overline{V}_c = V \setminus V_c$.

Let $\{V_c\}_{c=1}^{k}$ be a hard partitioning of G (i.e., $V_i \cap V_j = \emptyset$ if $i \neq j$, and $\cup_i V_i = V$), where V_c is the set of vertices in the cth

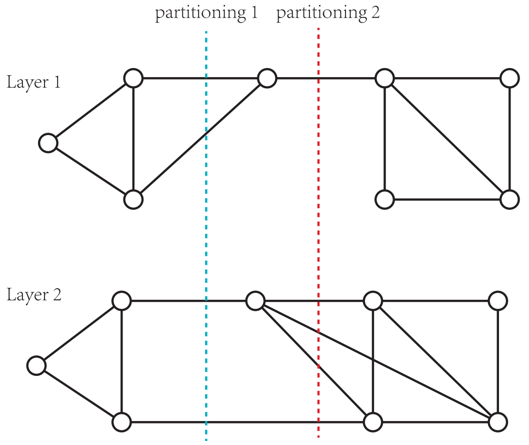

Fig. 1. The schematic example of communities in two-layer networks, where the partitioning is denoted by the two dashed lines. Partitioning 1 is better than partitioning 2 in the second layer network, whereas partitioning 2 is better than partitioning 1 in the first layer network. In all, partitioning 1 is better than partitioning 2 in the two-layer networks since there are fewer edges across communities in all layers.

community and k is the number of communities. The modularity density Q_D for $\{V_c\}_{c=1}^k$ is defined as [10]

$$Q_D(\{V_c\}_{c=1}^k) = \sum_{c=1}^k \frac{L(V_c,V_c) - L(V_c,\overline{V}_c)}{|V_c|}. \tag{1}$$

Multi-layer networks are defined as a sequence of single-layer networks, denoted by $\mathcal{G} = (G^{[1]}, \ldots, G^{[m]})$, where $G^{[l]} = (V^{[l]}, E^{[l]})$ is the lth layer network. Without loss of generality, we assume that networks have the same vertex set for each layer, i.e., $G^{[m]} = (V, E^{[m]})$. The weighted adjacency matrix for $\mathcal{G}$ is defined as $\mathcal{W} = [W^{[1]}, W^{[2]}, \ldots, W^{[m]}]$, where $W^{[l]}$ is the weighted adjacency matrix for $G^{[l]}$ and element $w_{ij}^{[l]}$ is the weight on edge (v_i, v_j) in $G^{[l]}$. The degree of vertex v_i in $G^{[l]}$ is defined as $d_i^{[l]} = \sum_j w_{ij}^{[l]}$. Similarly, the degree matrix for $\mathcal{G}$ is defined as $\mathcal{D} = [D^{[1]}, D^{[2]}, \ldots, D^{[m]}]$, where $D^{[l]}$ is the degree diagonal matrix for G_l, i.e., $D^{[l]} = diag\,(d_1^{[l]}, \ldots, d_n^{[l]})$.

To detect communities in multi-layer networks, a partitioning must be quantified by using the topology of all layers. A schematic example of communities in two-layer networks is shown in Fig. 1 with two partitionings (denoted by the dashed lines with various colors). Partitioning 1 (blue dashed line) is better than partitioning 2 (red dashed line) in the second layer because only 2 edges are cut by partitioning 1 and 4 edges are cut by partitioning 2. Partitioning 1 is worse than partitioning 2 in the first layer network because only 1 edge is cut by partitioning 2 and 2 edges are cut by partitioning 1. However, partitioning 1 is better than partitioning 2 in both two layers since fewer edges are cut by partitioning 1.

Ideally, we expect to obtain the optimal partitioning $\{V_c\}_{c=1}^k$ for a multi-layer network by maximizing the modularity density in each layer, i.e.,

$$\begin{cases} \max Q_D^{[1]}(\{V_c\}_{c=1}^k), \\ \max Q_D^{[2]}(\{V_c\}_{c=1}^k), \\ \cdots \\ \max Q_D^{[m]}(\{V_c\}_{c=1}^k), \end{cases} \tag{2}$$

where $Q_D^{[l]}(\{V_c\}_{c=1}^k)$ is the modularity density for $\{V_c\}_{c=1}^k$ in network $G^{[l]}$. Eq. (2) transforms the community detection in multi-layer networks into a multi-objective optimization problem. However, maximizing $Q_D^{[l]}$ for each layer is impossible. Thus, we relax the maximum for each layer by maximizing the average modularity density of multiple layers. We define the *multi-layer modularity density* of $\{V_c\}_{c=1}^k$ as

$$Q_D^{\mathcal{G}}(\{V_c\}_{c=1}^k) = \frac{1}{m}\sum_{l=1}^m Q_D^{[l]}(\{V_c\}_{c=1}^k). \tag{3}$$

The underlying assumption is that if a partitioning discovers well-connected subgraphs in all layers, then this partitioning receives a high value. One advantage of the multi-layer modularity density is that the multi-objective optimization problem in Eq. (2) is transformed into a single-objective optimization problem in Eq. (3). Similar to Q [9] and Q_D [10], $Q_D^{\mathcal{G}}$ achieves maximum when all vertices constitute a community. This apparently is trivial. However, valid communities can be obtained with $k > 1$.

Afterward, we prove the equivalence relation between $Q_D^{\mathcal{G}}$ and the objective functions of various algorithms for community detection in multi-layer networks.

4 Equivalence of Multi-Layer Modularity Density and Objectives of Algorithms

At first glance, the multi-layer modularity density and objective functions of these algorithms for community detection in multi-layer networks are unrelated. We prove the equivalence by making use of the trace optimization. Therefore, the key technique behind the theory is how to transform the objective functions into trace optimization.

4.1 Multi-Layer Modularity Density as Trace Maximization

The multi-layer modularity density is formulated as a trace optimization problem, which is summarized as the following:

Lemma 4.1. *Optimizing the multi-layer modularity density in Eq. (3) is proportional to the trace optimization of multi-layer networks, i.e.,* $\max Q_D^{\mathcal{G}}(\{V_c\}_{c=1}^k) \propto \max_{\tilde{X}} trace(\tilde{X}' \sum_{l=1}^m (2W^{[l]} - D^{[l]})\tilde{X})$.

Proof. Since X is the indication matrix for partitioning $\{V_c\}_{c=1}^k$, $\mathbf{x}_{\cdot c} = (x_{1c}, x_{2c}, \ldots, x_{nc})'$ is an indicator vector for the cth cluster V_c, where

$$x_{ic} = \begin{cases} 1, & \text{if } v_i \in V_c, \\ 0, & \text{otherwise.} \end{cases}$$

Thus, $\mathbf{x}'_{\cdot i}\mathbf{x}_{\cdot j} = |V_i|$ if $i = j$, 0 otherwise.

Then, $Q_D^{\mathcal{G}}$ in Eq. (3) can be re-written as

$$\begin{aligned} Q_D^{\mathcal{G}}(\{V_c\}_{c=1}^k) &= \frac{1}{m}\sum_{l=1}^m\sum_{c=1}^k \frac{L^{[l]}(V_c,V_c) - L^{[l]}(V_c,\overline{V}_c)}{|V_c|} \\ &= \frac{1}{m}\sum_{l=1}^m\sum_{c=1}^k \frac{2L^{[l]}(V_c,V_c) - L^{[l]}(V_c,V)}{|V_c|}. \end{aligned}$$

Because $L^{[l]}(V_c, V_c) = \mathbf{x}^{'}_{.c} W^{[l]} \mathbf{x}_{.c}$ and $L^{[l]}(V_c, V) = \mathbf{x}^{'}_{.c} D^{[l]} \mathbf{x}_c$, the above equation can be reformulated as

$$\begin{aligned} Q_D^{\mathcal{G}}(\{V_c\}_{c=1}^k) &= \frac{1}{m} \sum_{l=1}^m \sum_{c=1}^k \frac{\mathbf{x}^{'}_{.c} 2W^{[l]} \mathbf{x}_{.c} - \mathbf{x}^{'}_{.c} D^{[l]} \mathbf{x}_{.c}}{\mathbf{x}^{'}_{.c} \mathbf{x}_{.c}} \\ &= \frac{1}{m} \sum_{l=1}^m \sum_{c=1}^k \tilde{\mathbf{x}}^{'}_{.c} (2W^{[l]} - D^{[l]}) \tilde{\mathbf{x}}_{.c}, \end{aligned} \tag{4}$$

where $\tilde{\mathbf{x}}_{.c} = \mathbf{x}_c / (\mathbf{x}^{'}_{.c} \mathbf{x}_{.c})^{1/2}$.

Casting the last term of Eq. (4) as trace form, we obtain

$$\begin{aligned} Q_D^{\mathcal{G}}(\{V_c\}_{c=1}^k) &= \frac{1}{m} \sum_{l=1}^m trace(\tilde{X}^{'} (2W^{[l]} - D^{[l]}) \tilde{X}) \\ &= \frac{1}{m} trace(\tilde{X}^{'} \sum_{l=1}^m (2W^{[l]} - D^{[l]}) \tilde{X}), \end{aligned}$$

where $trace(W)$ is the trace of matrix W.

Thus, the assertion holds, i.e.,

$$\max Q_D^{\mathcal{G}} \propto \max_{\tilde{X}} trace\left(\tilde{X}^{'} \sum_{l=1}^m (2W^{[l]} - D^{[l]}) \tilde{X} \right). \tag{5}$$

□

4.2 Multi-Layer Modularity Density and Kernel K-Means

In single-layer networks, the equivalence between Q_D and kernel K-means is proved in Ref. [10]. Here, we extend the theory for multi-layer networks.

The objective function of kernel K-means is defined as [44]

$$F(\{V_c\}_{c=1}^k) = \sum_{c=1}^k \sum_{v_i \in V_c} \| \phi(v_i) - \mathbf{m}_c \|^2, \tag{6}$$

where $\mathbf{m}_c = \frac{\sum_{v_i \in V_c} \phi(v_i)}{|V_c|}$ is the center or mean of community V_c and $\phi(v_i)$ is a kernel function mapping vertex v_i onto a generally high dimensional space. Eq. (6) can also be transformed into a trace optimization problem as

Lemma 4.2 [44]. $F(\{V_c\}_{c=1}^k) = trace(\Phi^{'} \Phi) - trace(\tilde{X}^{'} \Phi^{'} \Phi \tilde{X})$, *where* $\Phi = (\phi(v_1), \phi(v_2), \ldots, \phi(v_n))$.

Theorem 4.3. *Optimizing the multi-layer modularity density is equivalent with optimizing the kernel K-mean algorithm for the multi-layer networks, i.e.,* $\max Q_D^{\mathcal{G}}(\{V_c\}_{c=1}^k) \propto \min F(\{V_c\}_{c=1}^k)$.

Proof. To proof the theorem, we need transform the objective function of the kernel K-means algorithm for multi-layer networks into a trace optimization problem.

Given multiple data (corresponding to the multi-layer networks), the kernel function for the lth data is denoted by $\Phi^{[l]}$. Therefore, the objective function of kernel K-means in Lemma 4.2 for multiple data can be reformulated as

$$\begin{aligned} F(\{V_c\}_{c=1}^k) &= \frac{1}{m} \sum_{l=1}^m trace((\Phi^{[l]})^{'} \Phi^{[l]}) \\ &- \frac{1}{m} \sum_{l=1}^m trace(\tilde{X}^{'} (\Phi^{[l]})^{'} \Phi^{[l]} \tilde{X}). \end{aligned} \tag{7}$$

Since the first term in the right part of Eq. (7) is a constant, minimizing $F(\{V_c\}_{c=1}^k)$ is equivalent to maximizing $trace(\tilde{X}^{'} (\Phi^{[l]})^{'} \Phi^{[l]} \tilde{X})$. Thus, the following expression holds

$$\min F(\{V_c\}_{c=1}^k) \propto \max_{\tilde{X}} trace\left(\tilde{X}^{'} \sum_{l=1}^m K^{[l]} \tilde{X} \right), \tag{8}$$

where $K^{[l]} = (\Phi^{[l]})^{'} \Phi^{[l]}$ is the kernel matrix for the lth type of data.

Comparing Eqs. (5), (6), (7), and (8), we can assert the equivalence between the objective of the kernel K-means and $Q_D^{\mathcal{G}}$ in multi-layer networks by setting $K^{[l]} = 2W^{[l]} - D^{[l]}$, i.e.,

$$\max Q_D^{\mathcal{G}}(\{V_c\}_{c=1}^k) \propto \max F(\{V_c\}_{c=1}^k). \tag{9}$$

□

Note that matrix $2W^{[l]} - D^{[l]}$ must be positive semi-definite to guarantee the convergence of kernel K-means. To solve this problem, a factor matrix is added to the kernel matrix as $K^{[l]} = \sigma I + 2W^{[l]} - D^{[l]}$, where σ is a real number chosen to be large enough so that $K^{[l]}$ is positive definite. Such an operator does not change the equivalence relation in Eq. (9), since the following equation holds

$$\begin{aligned} & trace\left(\tilde{X}^{'} \sum_{l=1}^m (\sigma I + 2W^{[l]} - D^{[l]}) \tilde{X} \right) \\ &= m\sigma + trace(\tilde{X}^{'} (2W^{[l]} - D^{[l]}) \tilde{X}). \end{aligned}$$

4.3 Multi-Layer Modularity Density and Nonnegative Matrix Factorization

NMF is an effective technique for dimension reduction [45], [46] and symmetric NMF (SNMF) is an extension [47]. In detail, given the data matrix $W_{n \times n}$ and number of communities k, SNMF approximately factorizes W into a matrix $B_{n \times k}$ such that

$$W \approx \tilde{X} \tilde{X}^{'}, s.t. \tilde{X} \geq 0, \tag{10}$$

where k is much smaller than n. The objective function of SNMF for multi-layer networks is defined as

$$O_{NMF}^{\mathcal{G}} = \sum_{l=1}^m \| W^{[l]} - \tilde{X} \tilde{X}^{'} \|^2 / m. \tag{11}$$

Theorem 4.4. *Optimizing the multi-layer modularity density is equivalent with SNMF for the multi-layer networks, i.e.,* $\max Q_D^{\mathcal{G}}(\{V_c\}_{c=1}^k) \propto \min_{\tilde{X} \geq 0} O_{NMF}^{\mathcal{G}}$.

Proof. To prove the equivalence between SNMF and $Q_D^{\mathcal{G}}$, we first transform the objective function of SNMF for multi-layer networks into trace optimization. Let $O_{NMF}^{[l]}$ be the objective of SNMF for the lth layer of $\mathcal{G}$, i.e.,

$$O_{NMF}^{[l]} = \min_{\tilde{X} \geq 0} \| W^{[l]} - \tilde{X} \tilde{X}^{'} \|^2 / m. \tag{12}$$

Since $\| W^{[l]} \|^2$ is a constant for a given layer $G^{[l]}$, maximizing modularity density for $G^{[l]}$ (i.e., $\max Q_D^{[l]}(\{V_c\}_{c=1}^k)$) is re-written as

$$\begin{aligned}
&\max_{\tilde{X}\geq 0, \tilde{X}'\tilde{X}=I_k} trace(\tilde{X}' W^{[l]} \tilde{X}) \\
&\propto - \min_{\tilde{X}\geq 0, \tilde{X}'\tilde{X}=I_k} trace(\tilde{X}' W^{[l]} \tilde{X}) \\
&\propto \min_{\tilde{X}\geq 0, \tilde{X}'\tilde{X}=I_k} \|W^{[l]}\|^2 - 2trace(\tilde{X}' W^{[l]} \tilde{X}) + \|\tilde{X}'\tilde{X}\|^2 \\
&= \min_{\tilde{X}\geq 0, \tilde{X}'\tilde{X}=I_k} \|W^{[l]} - \tilde{X}\tilde{X}'\|^2.
\end{aligned}$$

Relaxing the orthogonality $\tilde{X}^T\tilde{X} = I_k$, we can assert that SNMF is equivalent to $Q_D^{[l]}(\{V_c\}_{c=1}^k)$, i.e.,

$$\max Q_D^{[l]}(\{V_c\}_{c=1}^k) \propto \min_{\tilde{X}\geq 0} O_{NMF}^{[l]}. \tag{13}$$

Thus, we assert that the maximal multi-layer modularity density is equivalent with SNMF for multi-layer network, because the following equation holds

$$\max \sum_{l=1}^{m} Q_D^{[l]}(\{V_c\}_{c=1}^k)/m \propto \min_{\tilde{X}\geq 0} \sum_{l=1}^{m} \|W^{[l]} - \tilde{X}\tilde{X}'\|^2/m. \tag{14}$$

□

4.4 Multi-Layer Modularity Density and Spectral Clustering

Dillion et al. [44] proved the equivalence of the target functions of spectral clustering and K-means. Here, we derive the relation among the association [48], graph cut [49], [50] and $Q_D^{\mathcal{G}}$ in multi-layer networks. The spectral clustering based on association and graph cut can be expressed as trace optimization form:

Lemma 4.5. [48], [50]. The association and graph cut for single-layer network G_m can be expressed as trace maximization: (1) $Ass^{[l]}(\{V_c\}_{c=1}^k) = Trace(\tilde{X}' W^{[l]}\tilde{X})$ and (2) $Cut^{[l]}\ (\{V_c\}_{c=1}^k) = trace(\tilde{X}'(D^{[l]} - W^{[l]})\tilde{X})$.

We extend the association and graph cut of $\{V_c\}_{c=1}^k$ in multi-layer networks as

$$Ass^{\mathcal{G}}(\{V_c\}_{c=1}^k) = \frac{1}{m}\sum_{l=1}^{m} Ass^{[l]}(\{V_c\}_{c=1}^k), \tag{15}$$

and

$$Cut^{\mathcal{G}}(\{V_c\}_{c=1}^k) = \frac{1}{m}\sum_{l=1}^{m} Cut^{[l]}(\{V_c\}_{c=1}^k). \tag{16}$$

The equivalence between $Q_D^{\mathcal{G}}$ and spectral clustering is stated as

Theorem 4.6. *Optimizing the multi-layer modularity density is equivalent with spectral clustering for the multi-layer networks based on association and graph cut: (1)* $\max Q_D^{\mathcal{G}}(\{V_c\}_{c=1}^k) \propto Ass^{\mathcal{G}}(\{V_c\}_{c=1}^k)$, *(2)* $\max Q_D^{\mathcal{G}}(\{V_c\}_{c=1}^k) \propto Cut^{\mathcal{G}}(\{V_c\}_{c=1}^k)$.

Proof. (1) Based on Lemma 4.5 and Eq. (15), $Ass^{[\mathcal{G}]}$ can be transformed to trace maximization as

$$\max Ass^{\mathcal{G}}(\{V_c\}_{c=1}^k) = \frac{1}{m}\max_{\tilde{X}} trace\left(\tilde{X}'\sum_{l=1}^{m} W^{[l]}\tilde{X}\right). \tag{17}$$

Comparison between Eqs .(17) and (5) indicates that optimizing Ass on the $\mathcal{G}$ is equivalent to optimizing $Q_D^{\mathcal{G}}$ on $\sum_{l=1}^{m}(2W^{[l]} - D^{[l]})$, i.e.,

$$\max Q_D^{\mathcal{G}}(\{V_c\}_{c=1}^k) \propto \max Ass^{\mathcal{G}}(\{V_c\}_{c=1}^k). \tag{18}$$

(2) Analogously, we can prove the equivalence between $Q_D^{\mathcal{G}}$ and graph cut $Cut^{\mathcal{G}}$. □

4.5 Multi-Layer Modularity Density and Multi-View Clustering

In multi-view clustering algorithms, each layer of networks corresponds a *single view*, which look for clusterings that are consistent across the views. Here, we address the equivalence between the multi-view clustering [38], [39] and $Q_D^{\mathcal{G}}$.

In last sections, we extend the spectral clustering and NMF for the multi-layer networks, where these algorithms optimize directly multi-layer networks. However, the multi-view clusterings obtain communities in by optimizing the feature matrices associated with multi-layer networks, which is quite different from the multi-layer algorithms. Specifically, in the single layer network $G^{[l]}$, the spectral clustering is to solve the following optimization as shown in Lemma 4.5

$$\max_{X^{[l]}} Trace((X^{[l]})' W^{[l]} X^{[l]}). \tag{19}$$

The co-regularized multi-view spectral clustering [38] combines $X^{[1]}, \ldots, X^{[m]}$ to obtain the consensus matrix $\tilde{X}$ via regularization, where the objective function is formulated as trace optimization, i.e.,

$$O_{MVspec}^{\mathcal{G}} = Trace\left(X'\sum_{l=1}^{m} X^{[l]}(X^{[l]})' X\right). \tag{20}$$

Comparison between Eqs. (20) and (5) indicates the equivalent relation between $Q_D^{\mathcal{G}}$ and the co-regularized multi-view spectral clustering by setting $\sum_{l=1}^{m} W^{[l]} = \sum_{l=1}^{m} X^{[l]}(X^{[l]})'$, which is stated as

Theorem 4.7. Optimizing the multi-layer modularity density is equivalent with co-regularized multi-view spectral clustering, i.e., $\max Q_D^{\mathcal{G}}(\{V_c\}_{c=1}^k) \propto O_{MV\ spec}^{\mathcal{G}}$.

What we want to point out is that there are at least two advantages for the equivalence among various algorithms in multi-layer networks:

- First of all, the equivalence provides a generalized framework for the kernel K-means, NMF, spectral clustering and multiview clustering for community detection in multi-layer networks. Even though the mechanisms of these algorithms differ greatly, the objective functions can be transformed into a trace optimization. The equivalence provides the theoretical explanation why these methods are also applicable for community detection in multi-layer networks.
- The trace optimization of objective functions implies that various algorithms optimize various matrices. Thus, it is natural to incorporate priori information into the matrices without modifying the structure of algorithms, which serves as the theoretical foundation for the proposed algorithm.

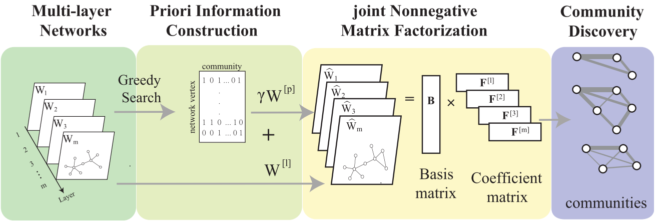

Fig. 2. Overview of the S2-jNMF algorithm, which comprises three major components, namely, priori information procedure which extracts the common dense subgraphs to construct the priori matrix, matrix decomposition procedure which extracts features by jointly factorizing the multiple matrices, and module discovery procedure which identifies communities in multi-layer networks based on the basis matrix.

Based on the proven equivalence, we design algorithms to discover communities in multi-layer networks in the next section.

5 Semi-Supervised Joint Nonnegative Matrix Factorization Algorithm (S2-jNMF)

The procedure of the proposed algorithm, the parameter selection, and the algorithm analysis are presented in this section.

5.1 Algorithm Procedure

As shown in Fig. 2, the S2-jNMF algorithm consists of three major components, namely, priori information construction, matrix factorization, and community discovery, which are discriminated by various background colors. To improve the performance of the algorithm, S2-jNMF incorporates priori information into the objective function, instead of directly factorizing the matrices associated with multi-layer networks. Since the ultimate goal is to discover common communities in multi-layer networks, the well-connected motifs (subgraphs) in all layers of networks are treated as priori information that is extracted using a greedy search method. Afterward, the algorithm jointly decomposes the matrices associated with multi-layer networks and priori information into a basis matrix and multiple coefficient matrices. The communities are discovered based on the common basis matrix.

5.1.1 Priori Information Construction

Unsupervised clustering methods imply that no outcome measure is available and that nothing is known about the relationship among the data. However, in many situations, one may wish to perform cluster analysis when an outcome variable exists or some preliminary information about the clusters is known. Some clues can also help us extract useful priori information. For example, in community detection, the dense subgraphs are very likely to be communities [14].

Furthermore, many algorithms have shown that properly incorporating priori information into clustering algorithms not only improves the accuracy of these algorithms but also increases their speed. For example, the pagerank algorithm demonstrates that even random partial information accelerates the convergence of an algorithm [51]. The semi-supervised NMF [14] and semi-supervised spectral clustering [55] outperform the conventional methods. However, to the best of our knowledge, no supervised-clustering- or semi-supervised-based algorithms for community detection in multi-layer networks are available. In this work, we present the first semi-supervised based algorithm that provides a general framework for community detection in multi-layer networks.

Priori information is constructed in two steps, namely, obtaining high-quality groups of vertices and constructing partial information. In the first step, we use a greedy search strategy to discover dense subgraphs in multi-layer networks. This strategy starts from a vertex (seed) and keeps expanding the subgraph until the density of the subgraph in some layers is less than a predefined threshold. The ultimate goal of the priori information construction is to identify subgraphs that are well connected in all layers. The strategy adopted for the priori information construction is based on seed-and-expand. We rank vertices is quantify centrality of vertices in multi-layer networks. The underlying assumption is that critical vertices are very likely to be the center of subgraphs.

To avoid a random selection of vertices, we select the unlabeled, high-ranking vertices as seeds. Therefore, we need to rank the vertices in multi-layer networks. We rank the vertices for each layer by using network propagation, which is also used for ranking disease genes [56]. Specifically, for each layer $G^{[l]}$, the vertices are ranked to construct a function $f^{[l]} \rightarrow R$, where $f^{[l]}(j)$ denotes the importance of vertex v_j in $G^{[l]}$. It is defined as

$$f^{[l]} = W^{[l]} f^{[l]}. \tag{21}$$

The above equation is solved by using an iterative strategy [57], i.e.,

$$f^{[l]}_{t+1} = W^{[l]} f^{[l]}_t, \tag{22}$$

where t is the number of iterations, which is usually 10. Then, we obtain the ranks of vertices in all layers as $[f^{[1]}, \ldots, f^{[m]}]$. Then, we rank the vertices in all layers by using the average score of the ranks.

To ensure the quality of the priori information, we use the greedy search algorithm to discover the community which maintains a strong connectivity in all layers. The greedy search starts from a vertex and keeps expanding the community by adding a vertex outside of the community. This procedure is repeated until the minimum density of the community in some networks is less than a predefined threshold β (how parameter β affects the performance of the algorithm will be discussed in the latter part of this paper).

The dense subgraphs are denoted by $\{C_i^{[p]}\}_{i=1}^{k'}$, where k' is the number of subgraphs detected by the greedy search algorithm. We construct a vector $\mathbf{z}_i$ for each subgraph $C_i^{[p]}$ as

$$z_i(j) = \begin{cases} 1, & \text{if } v_j \in C_i^{[p]}, \\ 0, & \text{otherwise}, \end{cases} \tag{23}$$

Based on the partial vectors, we construct the partial matrix as

$$W^{[p]} = \sum_{i=1}^{k'} \mathbf{z}_i \mathbf{z}_i'. \tag{24}$$

$W^{[p]}$ is a block-diagonal matrix with elements 0 and 1, and the $w_{ij}^{[p]}$=1 indicates the the two vertices are more likely to be clustered together. The ultimate goal of the priori matrix is to guide the community detection in multi-layer networks. Similar to the strategy in [14], [55], we incorporate the priori information into $W^{[l]}$ as

$$\widehat{W}^{[l]} = W^{[l]} + \gamma W^{[p]}, \tag{25}$$

where γ is a parameter that controls the relevant weight of the priori information (how parameter γ affects the performance of the algorithm will be discussed in the experiments).

5.1.2 Joint Nonnegative Matrix Factorization

Given $\widehat{W}^{[l]}(1 \leq l \leq m)$, the single-layer community detection algorithms apply the conventional NMF algorithm to each layer. However, this strategy ignores the connection among various layers, thereby resulting in an undesirable performance. To overcome this problem, we factorize $\widehat{W}^{[l]}(1 \leq l \leq m)$ by extending the NMF algorithm to jointly decompose these matrices into a common basis matrix B and various coefficient matrices $F^{[l]}(1 \leq l \leq m)$, i.e.,

$$\widehat{W}^{[l]} \approx BF^{[l]}, 1 \leq l \leq m. \tag{26}$$

Ideally, we can solve Eq. (26) by minimizing the reconstruction error, i.e.,

$$J = \sum_{l=1}^{m} \min \|\widehat{W}^{[l]} - BF^{[l]}\|^2 \tag{27}$$

$$s.t. \begin{cases} B \geq 0, \\ F^{[l]} \geq 0, \end{cases}$$

Since it is difficult to reach the minimum for each layer, we reformulate the above equation as

$$J = \min \sum_{l=1}^{m} \|\widehat{W}^{[l]} - BF^{[l]}\|^2 \tag{28}$$

$$s.t. \begin{cases} B \geq 0, \\ F^{[l]} \geq 0, \end{cases}$$

By setting

$$\begin{cases} \frac{\partial J}{\partial B} = 0, \\ \frac{\partial J}{\partial F^{[l]}} = 0, \end{cases}$$

we obtain the update rules for matrices B and $F^{[l]}$

$$B = B \frac{\sum_l \widehat{W}(F^{[l]})'}{B \sum_{l=1} F^{[l]}(F^{[l]})'}, \tag{29}$$

and

$$F^{[l]} = F^{[l]} \frac{B' \widehat{W}_l}{B' B F^{[l]}}. \tag{30}$$

The iteration is continued until the reconstruction error J is lower than a predefined tolerant error (e.g., 10^{-2}), or the maximum iteration number (e.g., 10^2) is reached.

5.1.3 Community Discovery

After obtaining the basis B, the indicator matrix X for the communities in multi-layer networks is constructed based on matrix B, i.e., $x_{ij^*} = 1$, where $j^* = \arg max_j B_{ij}$, 0 otherwise. S2-jNMF is presented in Algorithm 1.

Algorithm 1. The S2-jNMF Algorithm

```
Input:
    $\mathcal{G}$: Multi-layer networks;
    $k$: Number of communities;
    $\gamma$: Weight for priori information;
    $\beta$: Threshold of density for priori information;
Output:
    $\{C_i\}_{i=1}^{k}$: Communities in $\mathcal{G}$.
        Part I: Priori information
1: Obtaining the common dense subgraph in multi-layer
   network by using the greedy search;
2: Constructing the semi-supervised matrix $W^{[p]}$ as Eq. (24);
3: Constructing the matrix $\widehat{W}^{[l]}$ as Eq. (25);
        Part II: Matrix decomposition
4: Fixing coefficient matrix $F^{[l]}(1 \leq l \leq m)$, update the basis
   matrix $B$ as Eq. (29);
5: Fixing basis matrix $B$, update the coefficient matrix
   $F^{[l]}(1 \leq l \leq m)$ as Eq. (30);
6: Repeating the step 4 and 5 until the termination criterion is
   reached;
        Part III: Community discovery
7: Obtaining the module using the basis matrix $B$;
8: return $\{C_i\}_{i=1}^{k}$.
```

5.2 Parameter Selection

The S2-jNMF algorithm involves three parameters k, β and γ, where k is the number of communities, β is the density threshold for priori information, and γ controls the importance of priori information.

Determining the number of clusters is a challenging task in clustering that can be solved in many ways. For instance, in the spectral clustering algorithm, the gap between two successive eigenvalues is used to select the appropriate k. When the network size is small, a large gap can be easily obtained. When the size of the network is large, the gap is difficult to find because the threshold cannot be easily defined. The objective function based strategy is also commonly used, where k is the value corresponding to the maximum (minimum) value of a given objective function, such as modularity Q or modularity density Q_D. However, the resolution limit problem prefers to the communities with large sizes [52].

Wu et al. [53] recently proposed the instability-based NMF model for parameter selection, which combines multiple runs of NMF with a criterion to measure the instability of the output matrix. For each k, the NMF algorithm runs τ times with random initial solutions and obtains τ basis matrices, which are denoted by $B_1, \ldots, B_\tau$. Given two matrices B_1 and B_2, a $\tau \times \tau$ matrix H is defined where element h_{ij} denotes the cross correlation between the ith column of matrix B_1 and the jth column of matrix B_2. The dissimilarity between B_1 and B_2 is defined as

$$diss(B_1, B_2) = \frac{1}{2k}\left(2k - \sum_j \max H_{.j} - \sum_i \max H_{i.}\right),$$

where $H_{.j}$ denotes the jth column of matrix Q. The instability is the discrepancy of all the basis matrices for k, which is defined as

TABLE 2
The Statistics of Datasets

	$\vert V\vert$	$\vert E\vert$	m
Artificial networks # 1	1,024	327,680	5
Artificial networks # 2	1,024	327,680	5
Cell phone networks	400	10,500	10
Cancer networks	15,054	11,752,502	4
Amazon co-shopping networks	12,284	1,570,668	3

$$\Upsilon(k) = \frac{2}{\tau(\tau-1)} \sum_{1\le i<j\le\tau} diss(B_i, B_j).$$

The k corresponding to the minimal $\Upsilon(k)$ is selected as the number of communities. We set γ and β as the tuning parameters, which are selected empirically.

5.3 Algorithm Analysis

We first discuss the space complexity of S2-NMF. Given the multi-layer network $\mathcal{G}$, the 3-dimensional adjacency matrix $\mathcal{W}$ requires space $O(n^2m)$. The space for factorizing the W is $O(nk(m+1))$ to store the basis matrix B and coefficient matrix $F^{[l]}(1\le l\le m)$. Therefore, the overall space complexity of S2-jNMF is $O(n^2m)$.

Afterward, we analyze the time complexity. S2-jNMF consists of three major components: the priori information construction by using greedy search, matrix factorization, and module discovery. The time complexity of the greedy search is $O(mn^2)$, which that for jNMF is $O(mrn^2k)$, where r is the number of iterations [54]. Thus, the overall time complexity of the proposed algorithm is $O(rn^2km)$, showing that the proposed method does not increase the time complexity compared to NMF.

The complexity of S2-jNMF can be improved. Specifically, the space complexity of the proposed algorithm can be significantly reduced because most of networks are sparse, i.e., $O(|E^{[l]}|)\ll O(n^2)$. Thus, we can only put the edge information into memory, where each edge is represented by a triplet $\{i, j, w_{ij}^{[l]}\}$. Therefore, for each layer the space complexity of the proposed algorithm is $O(\sum_{l=1}^{m}|E^{[l]}|)$. Other strategies can also accelerate the S2-jNMF algorithm via regularization. We demonstrate that the sparseness regularization saves about 10 percent of the running time without sacrificing accuracy (supplementary materials, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TKDE.2018.2832205).

6 Experiments

To fully evaluate the performance of the S2-jNMF algorithm, six well-known algorithms are adapted for comparison, including MVspec [38], MVnmf [39], M-Module [29], MolTi [40], spectral clustering [43], and NMF [45],. Notice that both spectral clustering and NMF are not suitable for multi-layer networks, and we only extend them based on consensus clustering strategy [32] (denoted by conSPEC and conNMF). All these algorithms can be classified into three classes: topology-, consensus-, and multi-view-based approaches. The topology-based algorithms include M-Module and MolTi, which cluster multi-layer networks by exploiting the topological structure during the module search procedure. The consensus-based algorithms include conNMF and conSPEC, which use traditional approaches to identify the modules in each layer independently and then obtain the communities via consensus clustering. The multi-view-based algorithms include MVspec and MVnmf.

Five datasets, including two artificial and three real-world networks, are used to validate the performance of the aforementioned algorithms. These datasets are summarized in Table 2. The artificial networks are used for measuring the accuracy of various algorithms, while the real-world networks are used to test whether these algorithms can discover communities in multi-layer networks with specific backgrounds.

6.1 Settings

If the community structure is known, then we adopt the normalized mutual information (NMI) [15] to measure the performance of algorithms because the community structure are known. Specifically, given the standard partitioning $\mathcal{C}^*$and obtained partitioning $\mathcal{C}$, we construct a confusion matrix N whose rows and columns correspond to the communities in $\mathcal{C}^*$ and $\mathcal{C}$, respectively. Element N_{ij} is the number of vertices overlapped by the ith real and jth obtained communities. NMI is defined as

$$NMI(\mathcal{C}^*,\mathcal{C}) = \frac{-2\sum_{i=1}^{|\mathcal{C}|}\sum_{j=1}^{|\mathcal{C}^*|} N_{ij}\log\frac{N_{ij}N}{N_{i.}N_{.j}}}{\sum_{i=1}^{|\mathcal{C}|} N_{i.}\log\frac{N_{i.}}{N} + \sum_{j=1}^{|\mathcal{C}^*|} N_{.j}\log\frac{N_{.j}}{N}},$$

where $|\mathcal{C}|$ is the number of communities in $\mathcal{C}$, and $N_{i.}$ is the sum of the ith row of N.

For the multi-layer networks whose community structure is unknown, we adopt density and association Ass (Lemma 4.5) to measure the performance of algorithms. We compare the distributions of the average density and ass of modules across all layers to measure the performance of these algorithms.

6.2 Artificial Networks #1

The first dataset is based on GN benchmark networks [1], which consist of 1,024 vertices grouped into 16 communities that contains 64 vertices each. Every vertex has an average degree of 64 and shares Z_{out} edges that connect the vertices outside of the corresponding community. The parameter Z_{out} controls the noise level. We construct five-layer networks with various Z_{out}. To fully validate the performance of the compared algorithms, two types of multi-layer networks are constructed: *HOMO* and *HETER* networks. The parameter Z_{out} is the same for all layers in HOMO networks, but varies across all layers in HETER networks. Specifically, parameter Z_{out} increases from 4 to 32 in HOMO, but is fixed to 8 in the first two layers and varies from 4 to 32 for the other three layers. The heatmap of subgraphs of five-layer HETER networks is shown in Fig. 3 A, where Z_{out} is fixed to 8 in the first two layers, and to 24 for the other three layers. The vertices surrounded by the dashed lines denote the communities in five-layer networks. The accuracy of these algorithms is obtained as the average NMI over 50 runs.

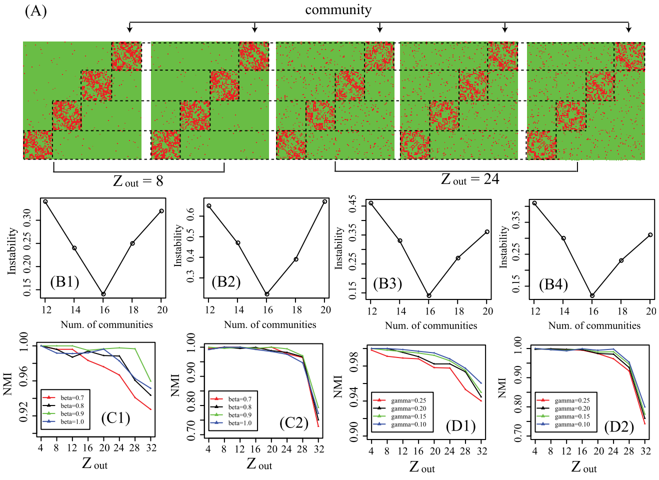

Fig. 3. Parameter effects. (A) heatmap of five-layer HETER networks, where Z_{out}=8 for the first two layers and 24 for the other three layers. The communities are surrounded by dashed lines. (B) Instability of S2-jNMF for selecting the number of communities on various multi-layer networks: (B1) HOMO networks with Z_{out}=4, (B2) HOMO networks with Z_{out}=12, (B3) HETER networks with Z_{out}=4, and (B4) HETER networks with Z_{out}=12. (C) Parameter effects of β on different multi-layer networks: (C1) HOMO networks and (C2) HETER networks. (D) Parameter effects of γ on various multi-layer networks: (D1) HOMO networks and (D2) HETER networks.

Before assessing the performance of these algorithms, we investigate how the parameters affect the performance of S2-jNMF. Recall that three parameters are involved in S2-jNMF, where k is the number of communities, γ is the relative weight for the priori information, and β controls the density of subnetworks for the priori information. We analyze how each parameter affects the performance of S2-jNMF by fixing the other parameters. To investigate whether the instability-based strategy can precisely select the number of communities in multi-layer networks, we apply the S2-jNMF algorithm to the HOMO and HETER networks by varying k from 12 to 20 while fixingβ=0.9 and γ=0.1. The instability of S2-jNMF on various k is shown in Fig. 3B, where Fig. 3B1/B2 shows the results for the HOMO networks with Z_{out}=24/32, respectively. Fig. 3B3/B4 shows the results for the HETER networks with Z_{out}=24/32, respectively. Fig. 3B shows that S2-jNMF can accurately identify the optimal value of k because the instability reaches the minimum at k=16.

Afterward, we investigate how parameter β influences the performance of the S2-jNMF algorithm by adjusting γ=0.1. Fig. 3 shows how the NMI changes as β increases from 0.7 to 1 with a gap of 0.1, where Figs. 3C1 and 3C2 contain the NMIs for HOMO HETER networks, respectively. As β increases, the accuracy increases dramatically at first, and then decreases when $\beta > 0.9$. Such tendency can be attributed to the fact that when β is small, the communities obtained by the greedy search algorithm have a weak connectivity, thereby introducing noise in the priori information. Increasing β filters such noise, and consequently improves the accuracy of the algorithm. When β is large, the number of vertices for the priori information decreases, thereby reducing the accuracy. Therefore, β=0.9 reaches a balance between the quality of the priori information and the size of vertices for the priori information. In the following experiments, we set β=0.9.

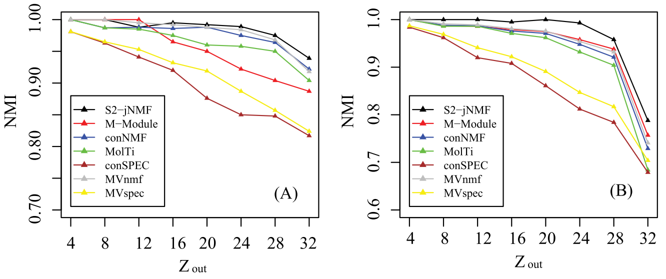

Fig. 4. Comparison among various algorithms in terms of NMI as parameter Z_{out} increases from 4 to 32 in: (A) HOMO networks and (B) HETER networks.

The impact of parameter γ on the S2-jNMF algorithm is shown in Fig. 3D with β=0.9, where Figs. 3D1 and 3D2 are the accuracy of S2-jNMF for HOMO and HETER networks, respectively. As γ increases, the accuracy dramatically decreases. This finding can be attributed to the fact that if γ is small, then the priori information is auxiliary. When γ is large, the contribution of the priori information to the accuracy increases, thereby resulting in an undesirable performance because the priori information deviates the direction of community search from the original networks. Fig. 3D suggests that γ=0.1 reaches a good balance between the priori information and original data, which is consistent with the assertion in Refs. [47], [55]. In the following experiments, we set γ=0.1.

The performance of the compared algorithms is presented in Fig. 4, where Figs. 4A and 4B contain the NMIs for various algorithms on HOMO and HETER networks, respectively. From Fig. 4, we conclude that as Z_{out} increases, the accuracy of algorithms decreases dramatically. The reason is that as Z_{out} increases, the community structure becomes fuzzy because more edges appear across modules. The spectral clustering algorithm has been widely studied used for community detection in single-layer networks because of the excellent performance [11], [12], [55]. Surprisely, conSPEC and MVspec show the worst performance among all compared algorithms, thereby indicating that the spectral information cannot sufficiently characterize the topological structure of multi-layer networks. MVnmf is inferior to S2-jNMF, yet outperforms the other algorithms. Furthermore, the S2-jNMF algorithm achieves the best performance, indicating the superiority of the proposed method. This result can be ascribed to two reasons. First, S2-jNMF jointly decomposes multi-layer networks, and simultaneously makes use of topological information of multiple layers during the module search procedure, whereas the consensus-clustering-based algorithms, such as conNMF and conSPEC, independently search the modules in each layer. Second, the appropriate semi-supervision strategy enhances the accuracy of the proposed algorithm. Taking these results together, we assert that the joint analysis of multi-layer networks is more promising than consensus and multi-view clustering.

A careful comparison of Figs. 4A and 4B demonstrates that the aforementioned tendency is similar for both the HOMO and HETER networks.

6.3 Artificial Networks #2

Since the artificial networks in dataset #1 are insufficient to fully validate the performance of the compared algorithms,

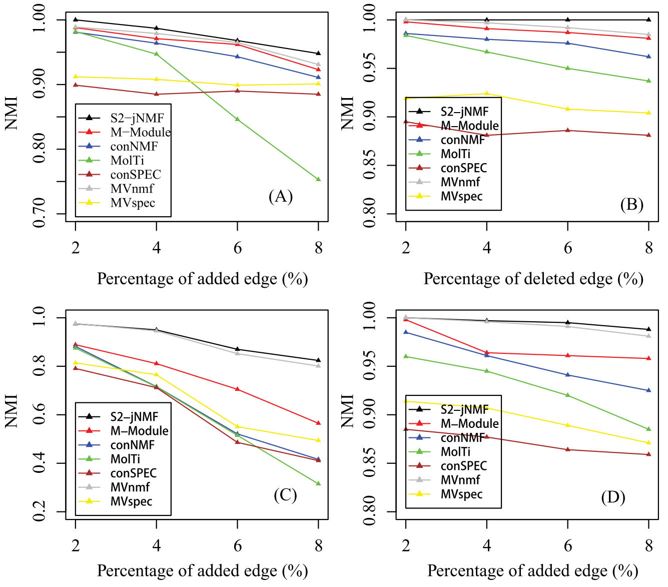

Fig. 5. Comparison of various algorithms in terms of NMI by adding or deleting t% of edges: (A) *ADD-Net* networks with Z_{out}= 20, (B) *DEL-Net* networks with Z_{out}=20, (C) *ADD-Net* networks with Z_{out}= 24, and (D) *DEL-Net* networks with Z_{out}= 24.

we construct more complicated multi-layer networks with different layers. In order to introduce dynamics in the multi-layer networks, for each layer, t% of edges are randomly added or deleted from the previous layer. To this end, two cases have been generated, namely, *ADD-Net* and *DEL-Net*. In the former, t% of edges are randomly added in the current layer. In the latter, t% of edges in the previous layer are randomly deleted to construct the current layer. To fully explore the performance of the compared algorithms, we vary t from 2 to 8. We consider five layers and obtain the accuracy of the algorithms as the average NMIs over 50 runs.

By setting Z_{out}=20, we generate two types of artificial multi-layer networks. The accuracy of various algorithms on the artificial networks is shown in Fig. 5, where Figs. 5A and 5B are the accuracy of algorithms on the ADD-Net networks and DEL-Net networks. From Fig. 5, as t increases from 2 to 8, the accuracy of algorithms decreases dramatically for both ADD-Net and DEL-Net networks. The reason is that the larger t is, the more edges are deleted or added. Moreover, adding or deleting edges changes the structure of communities, thereby increasing the difficulty to detect them. Furthermore, the S2-jNMF algorithm is more accurate than the other algorithms, thereby demonstrating that the S2-jNMF algorithm is more robust than the others. Consistent with dataset #1, the conSPEC and MVspec algorithms are worse than the others because spectrum is sensitive to network perturbation [12].

To further validate the performance of these algorithms, we apply them on artificial networks by setting Z_{out}=24 as shown in Figs. 5C and 5D. Both of these figures show that the similar tendency occurs. A careful comparison between the ADD-Net and DEL-Net networks indicates that algorithms demonstrate a better performance on DEL-Net networks than on ADD-Net networks, because deleting edges is more likely to preserve the community structure than adding edges.

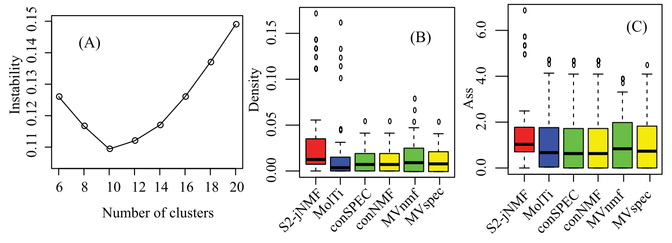

Fig. 6. Comparison of the performance of various algorithms on the cellphone networks: (A) Instability of S2-jNMF, (B) Distributions of the density of communities obtained by various algorithms, and (C) Distributions of the Ass of communities obtained by various algorithms.

6.4 Cellphone Networks

The previous two datasets are artificial networks where the membership of the vertices is known. We then check how the compared algorithms discover communities in real world multi-layer networks. The cellphone calls[1] among the members of the fictitious Paraiso movement are used. These phone calls have been conducted within a span of 10 days in June 2006. The cellphone network is constructed by regarding each individual as a vertex and regarding a call between a pair of members as an edge. For each day, a cellphone network is constructed. The total number of cellphones is 400.

Given that the true community structure is unknown, we use density and Ass to measure the performance of algorithms. The S2-jNMF discovers 10 communities in the cellphones as shown in Fig. 6A, where the instability reaches the minimum at k=10. The density of communities detected by various algorithms is shown in Fig. 6B, where the density of communities obtained by the S2-jNMF algorithm is much higher than that of the communities obtained by the other algorithms. Algorithms based on multi-view and consensus clustering show the same performance. The similar tendency repeats for Ass as shown in Fig. 6C. The results demonstrate that the S2-jNMF algorithm is much better than the others in real social multi-layer networks. The density of the obtained communities is low because the cellphone network is sparse.

6.5 Cancer Networks

The sizes of all previous multi-layer networks are too small to investigate the effectiveness of the compared algorithms. Therefore, in the following experiments, we use large-scale multi-layer networks to investigate the capability of algorithms in discovering communities. The first large-scale multi-layer networks is associated with breast cancer progression based on the gene expression data [29]. These networks have four layers with each layer corresponding to a specific clinical stage of breast cancer. The multi-layer networks comprise 15,054 genes and more than 11 millions edges. In the following experiments, the M-Module algorithm is excluded for fair comparison because this algorithm is not based on partitioning.

Density and Ass are used to quantify the performance of the compared algorithms because the benchmark community structure is unknown. The density of communities obtained by various algorithms is shown in Fig. 7A, where the S2-jNMF algorithm significantly outperforms the others except for MVnmf. Specifically, the average density of

[1] http://www.cs.umd.edu/hcil/VASTchange08/

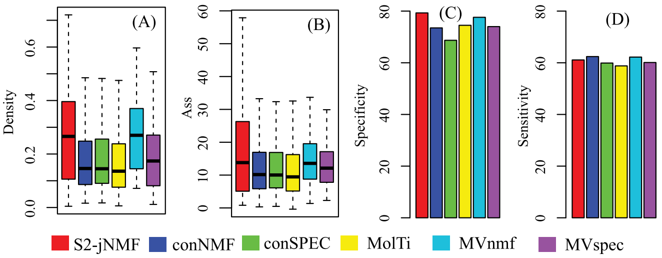

Fig. 7. Performance of various algorithms on cancer networks: (A) Distributions of the density of communities obtained by various algorithms, (B) Distributions of the Ass of communities obtained by various algorithms, (C) Specificity of modules obtained by various algorithms in the GO term enrichment analysis, and (D) Sensitivity of communities obtained by various algorithms in the GO term enrichment analysis.

modules obtained by S2-jNMF is 0.27, whereas that for conNMF (conSPEC, MolTi) is 0.17 (0.17, 0.16). Furthermore, the average Ass of the communities obtained by S2-jNMF is 14.2, while that of the communities obtained by conNMF (conSPEC, MolTi) is 9.9 (9.8, 9.5). These results further demonstrate that the S2-jNMF algorithm is effective in discovering communities in cancer networks. S2-jNMF and MVnmf show a similar performance because the latent space may capture the community structure of cancer networks.

Moreover, we check whether the obtained communities capture the biological backgrounds. We evaluate the obtained communities by using Gene Ontology [58] with an immediate purpose to check how well the discovered communities enriched the known biological functions. To evaluate the statistical and biological significance of the communities, the P-value [59] is employed. Specifically, given a community C with k genes in a functional group F, the P-value is defined as

$$P-value = 1-\sum_{i=0}^{k-1}\frac{\binom{|F|}{i}\binom{|V|-|F|}{|C|-i}}{\binom{|V|}{|C|}},$$

where $|V|$ denotes the number of genes in the network. The functional homogeneity P-value represents the probability for a given set of genes to be enriched by a function merely by chance. All P-values are corrected by the Benjamini-Hochberg method [60] with a cutoff of 0.05. Following the strategy in [29], we employ specificity and sensitivity as criteria to evaluate the performance of the algorithms. Specificity is defined as the percentage of enriched communities, i.e.,

$$specificity = \frac{k^*}{k}\times 100\%,$$

where k^* is the number of communities that are significantly enriched with at least one biological process. Similarly, sensitivity is defined as the percentage of enriched biological processes, i.e.,

$$sensitivity = \frac{n^*}{n}\times 100\%,$$

where n^* is the number of the biological processes that are significantly enriched with at least one community, and n is the number of biological processes.

Figs. 7C and 7D report specificity and sensitivity of communities obtained by various methods. S2-jNMF outperforms the other algorithms in terms of specificity, indicating that the proposed algorithm discovers the biological processes. For example, the specificity of S2-jNMF is 79.3 percent for GO terms, while conNMF, conSPEC, MolTi, MVnmf and MVspec have specifities of 73.5, 68.7, 74.5, 77.6, and 74.0 percent, respectively. On the sensitivity (Fig. 7D), they have similar performance. Generally, S2-jNMF achieves the best performance, indicating that S2-jNMF provides an effective way to explore biological multi-layer networks associated with cancers.

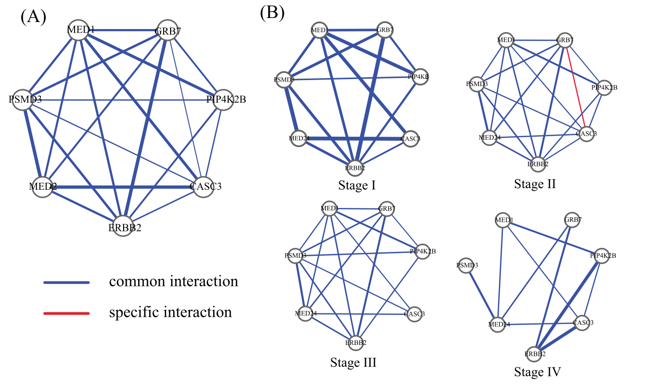

Fig. 8. A schematic community detected by S2-jNMF: (A) community represented by single-layer-based network, and (B) the topological structure of the community in multi-layer networks associated with various stages.

To demonstrate the superiority of S2-jNMF, a schematic example is shown in Fig. 8B. The community is significantly enriched by the HER2 pathway (P-value=4.7E-4), which is critical for breast cancer metastasis [61]. Based on single-layer-network-based community detection algorithms, we either cannot discover or identify the community structure as shown in Fig. 8A. The figure also demonstrates that we cannot determine which links are stage specific and which links are common. Furthermore, we do not know how the strength of links (edge weights) dynamically changes along with cancer progression. From Fig. 8B, we only not only identify the stage-specific and common links, but also obtain the dynamics of interactions.

6.6 Amazon Co-Shopping Networks

Amazon shopping data are used to test the performance of algorithms. These data collected by crawling the amazon website and contain product metadata and review information of 548,552 different products (books, music CDs, DVDs, and VHS videotapes) [62]. If two customers purchase the same product, then there is an edge between them. We construct three-layer networks for books, music CDs and DVDs, respectively. The conNMF, conSPEC and MolTi algorithms are excluded in this experiment because they are inferior to S2-jNMF and multi-view clustering.

The performance of various algorithms is shown in Fig. 9, where Figs. 9A and 9B are the distributions of density and Ass, respectively. From Fig. 9, we conclude that S2-jNMF outperforms MVnmf, but is inferior to MVspec. Furthermore, MVspec achieves the best performance in terms of density, whereas obtains the worst performance in terms of Ass. The reason is that the size of communities obtained by MVspec are much smaller than that of MVnmf and S2-jNMF, thereby leading to high density and low Ass.

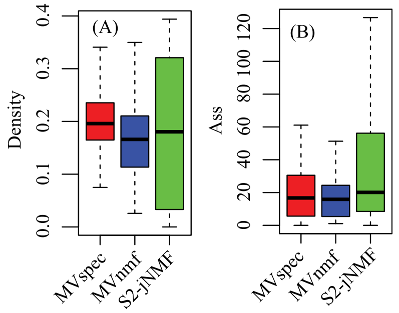

Fig. 9. Performance of various algorithms on amazon dataset: (A) Distributions of density of communities, and (B) distributions of Ass of communities.

The results further demonstrate that the proposed algorithm is promising in detecting multi-layer communities.

7 Conclusion

Complex systems are represented as networks that comprise multiple types of interactions that are organized as edges belonging to different layers. Investigating the communities in multi-layer networks has considerable merits because they provide insights into the structure of complex networks. For example, the communities within cancer networks provide value merits for the mechanisms of complex diseases. Although great efforts have been devoted to community detection in multi-layer networks, many problems remain unsolved. In this paper, we ascribe these problems to two issues. First, we propose a quantitative function for the communities for weighted and unweighted multi-layer networks. Second, we prove the equivalence among the kernel K-mean, spectral clustering, NMF, multi-view clustering, and the proposed quantification function. Furthermore, a semi-supervised joint nonnegative matrix factorization for detecting communities in multi-layer networks is proposed by incorporating partial information into NMF.

The following problems warrant for further research in future studies:

- In this study, we focus on community detection in multi-layer networks. Given that many other multi-layer networks, such as temporal, multiplex and interconnected networks, are available, how to precisely quantify communities in these networks and designing the appropriate algorithms for community detection would be interesting.
- Even though the time complexity of S2-jNMF is equal to that of the traditional NMF methods, it is still unacceptable for large-scale networks. How to accelerate the algorithms via network reduction, i.e., reducing the number of vertices in the network to preserve the patterns, is critical. Network embedding techniques [63], [64] have been extensively studied to analyze large-scale networks. How to use graph embedding to characterize and extract communities in multi-layer networks is also interesting.

Acknowledgments

This work was supported by the NSFC (Grant No. 61772394, 61502363, 61572385), International Cooperation and Exchange of the NSFC (Grant No. 61711530248), Science & Technology Program of Shannxi Province (Grant No. 2015KTCXSF-01), and Fundamental Research Funding of Central Universities (Grant no. JB180304). The authors appreciate the reviewers for their helpful suggestions.

References

[1] M. Girvan and M. E. J. Newmann, "Community structure in social and biological networks," *Proc. Nat. Academy Sci. United States America*, vol. 99, no. 12, pp. 7821–7826, 2002.

[2] A. Clauset, C. Moore, and M. E. J. Newman, "Hierarchical structure and the prediction of missing links in networks," *Nature*, vol. 453, no. 7191, pp. 98–101, 2008.

[3] J. Menche, A. Sharma, M. Kitsak, et al., "Uncovering disease-disease relationships through the incomplete interactome," *Sci.*, vol. 347, no. 6624, pp. 841–850, 2015.

[4] F. Zhang, W. Zhang, Y. Zhang, et al., "OLAK: An efficient algorithm to prevent unraveling in social networks," *Proc. VLDB Endowment*, vol. 10, no. 6, pp. 649–660, 2017.

[5] X. Wang, Y. Zhang, W. Zhang, and X. Lin, "Efficient distance-aware influence maximization in Geo-social networks," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 3, pp. 599–612, Mar. 2017.

[6] W. Liu, Z. Deng, X. Gong, et al., "Effectively predicting whether and when a topic will become prevalent in a social network," *Proc. 29th AAAI Conf. Artif. Intell.*, 2015, pp. 210–216.

[7] W. Liu, Z. Deng, L. Cao, et al., "Mining top K spread sources from a specific topic and a given node," *IEEE Trans. Cybern.*, vol. 45, no. 11, pp. 2472–2483, Nov. 2015.

[8] A. H. Tong, B. Drees, G. Nardelli, et al.,"A combined experimental and computational strategy to define protein interaction networks for peptide recognition modules," *Sci.*, vol. 295, pp. 321–324, 2002.

[9] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Phys. Rev. E*, vol. 69, 2004, Art. no. 026113.

[10] Z. P. Li, S. H. Zhang, and R. S. Wang, "Quantative function for community detection," *Phys. Rev. E*, vol. 77, 2008, Art. no. 036109.

[11] X. Ma, L. Gao, and X. Yong, "Eigenspaces of networks reveal the overlapping and hierarchical community structure more precisely," *J. Statistical Mech.: Theory Exp.*, vol. 08, 2010, Art. no. P08012.

[12] X. Ma and L. Gao, "Non-traditional spectral clustering algorithms for the detection of community structure in complex networks: A comparative analysis," *J. Statistical Mech.: Theory Exp.*, vol. 05, 2011, Art. no. P05012.

[13] X. Ma, B. Wang, and L. Yu, "Semi-supervised spectral algorithms for community detection in complex networks based on equivalence of clustering methods," *Physica A*, vol. 490, pp. 786–802, 2018.

[14] X. Ma, L. Gao, X. Yong, et al., "Semi-supervised clustering algorithm for community structure detection in complex networks," *Phys. A*, vol. 389, no. 1, pp. 187–197, 2010.

[15] L. Danon, J. Duch, A. Diaz-Guileram, and A. Arenas, "Comparing community structure identification," *J. Statistical Mech.*, vol. 2005, 2005, Art. no. P09008.

[16] A. Arenas, A. Fernandez, and S. Gomez, "Analysis of the structure of complex networks at different resolution level," *New J. Physics*, vol. 10, no. 5, pp. 4656–4658, 2008.

[17] X. Zhang and M. E. J. Newman, "Multiway spectral community detection in networks," *Phys. Rev. E*, vol. 92, 2015, Art. no. 052808.

[18] S. Fortunato, "Community detection in graphs," *Physics Rep.*, vol. 486, no. 3, pp. 75–174, 2009.

[19] M. Kivela, A. Arenas, M. Barthelemy, et al.,"Multilayer networks," *J. Complex Netw.*, vol. 2, pp. 203–271, 2014.

[20] R. Sanchez-Garcia , E. Cozzo, and Y. Moreno, "Dimensionality reduction and spectral properties of multilayer networks," *Phys. Rev. E*, vol. 89, 2014, Art. no. 052815.

[21] M. D. Domenico, V. Nicosia, A. Arenas, and V. Latora, "Structural reducibility of multilayer networks," *Nature Commun.*, vol. 6, 2015, Art. no. 6864.

[22] Q. Jiang, A. Campbell, G. Tang, and J. Pei, "Multi-level relationship outlier detection," *Int. J. Bus. Intell. Data Mining*, vol. 7, no. 4, pp. 253–273, 2012.

[23] Y. Wang, J. Pei, X. Lin, and Q. Zhang, "An iterative fusion approach to graph-based semi-supervised learning from multiple views," in *Proc. Pacific-Asia Conf. Knowl. Discovery Data Mining*, 2014, pp. 162–173.

[24] S. Boccaletti, G. Bianconi, et al.,"The structure and dynamics of multilayer networks," *Physics Rep.*, vol. 514, pp. 1–122, 2014.

[25] X. Ma, P. Sun, and Q. Gui, "Condition-specific modules by clustering multiple networks," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, 2017, to be published, doi: 10.1109/TCBB.2017.2761339.

[26] J. F. Padgett and C. K. Ansell, "Robust action and the rise of the Medici," *Amer. J. Sociology*, vol. 98, no. 6, pp. 1259–1319, 1992.

[27] A. Chatr-Aryamontri, B. J. Breitkreutz, R. Oughtred, et al.,"The BioGRID interaction database: 2015 update," *Nucleic Acids Res.*, vol. 43, pp. D470–D478, 2015.

[28] M. Szell, R. Lambiotte, and S. Thurner, "Multirelational organization of large-scale social networks in an online world," *Proc. Nat. Academy Sci. United States America*, vol. 107, no. 31, pp. 13636–13641, 2010.

[29] X. K. Ma, L. Gao, and K. Tan, "Modeling disease progression using dynamics of pathway connectivity," *Bioinf.*, vol. 36, no. 16, pp. 2343–2350, 2014.

[30] P. J. Mucha, T. Richardson, K. Macon, et al., "Community structure in time-dependent, multiscale, and multiple netowrks," *Sci.*, vol. 328, no. 5980, pp. 876–878, 2009.

[31] M. A. Rodriguez and J. Shinavier, "Exposing multi-relational networks to single-relational network analysis algorithms," *J. Informetrics*, vol. 4, no. 1, pp. 29–41, 2009.

[32] A. Lancichinetti and S. Fortunato, "Consensus clustering in complex networks," *Sci. Rep.*, vol. 2, 2012, Art. no. 336.

[33] S. Bickel and T. Scheffer, "Multi-view clustering," in *Proc. IEEE Int. Conf. Data Mining*, 2004, pp. 19–24.

[34] L. Tang, X. Wang, and T. Scheffer, "Community detection in multidimensional networks," Arizona State Univ., Tempe, AZ, USA, 2010.

[35] K. Chaudhuri, S. Kakade, et al., "Muti-view clustering via canonical correlation analysis," in *Proc. Annu. Int. Conf. Mach. Learn.*, 2009, pp. 129–136.

[36] A. Kumar and H. Daumé III, "A co-training approach for multiview spectral clustering," in *Proc. Annu. Int. Conf. Mach. Learn.*, 2011, pp. 393–400.

[37] Y. Fang, H. Zhang, Y. Ye, and X. Li, "Detecting hot topics from Twitter: A multiview approach," *Inf. Sci.*, vol. 40, no. 5, pp. 578–593, 2014.

[38] A. Kumar, P. Rai, and H. Daumé III, "Co-regularized multi-view spectral clustering," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2011, pp. 1413–1421.

[39] J. Liu, C. Wang, J. Gao, and J. Han, "Multi-view clustering via joint nonnegative matrix factorization," in *Proc. SIAM Data Mining Cnof.*, 2013, pp. 252–260.

[40] G. Didier, C. Brun, and A. Baudot, "Identifying communities from multiplex biological network," *Peer J.*, vol. 3, 2015, Art. no. 1525.

[41] T. P. Didier, "Inferring the mesoscale structure of layered, edge-valued, and time-varying networks," *Phy. Rev. E*, vol. 92, no. 4, 2015, Art. no. 042807.

[42] S. Paul and Y. Chen, "Null models and modularity based community detection in multi-layer networks," arXiv:1608.00623, 2016.

[43] M. E. J. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Phys. Rev. E*, vol. 74, 2006, Art. no. 036104.

[44] I. S. Dhillon, Y. Guan, and B. Kulis, "Weighted graph cuts without eigenvectors: A multilevel approach," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 11, pp. 1944–1957, Nov. 2007.

[45] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[46] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," *Auxiliary Commun. Navigat. Identification Panel*, vol. 32, no. 6, pp. 556–562, 2001.

[47] X. Ma, L. Gao, L. Fu, and X. Yong, "Semi-supervised clustering algorithm for community structure detection in complex networks," *Physica A*, vol. 389, no. 1, pp. 187–197, 2010.

[48] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[49] P. Chan, M. Schlag, and J. Zien, "Spectral K-way ratio cut partitioning and clustering," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 13, no. 9, pp. 1088–1096, Sep. 1994.

[50] S. X. Yu and J. Shi, "Multiclass spectral clustering," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2003, pp. 313–319.

[51] L. Page, S. Brin, R. Motwani, et al., "The PageRank citation ranking: Bringing order to the web," Tech. Rep. 1999-66, Standford Info Lab, 1999.

[52] S. Fortunato and M. Barthélemy, "Resolution limit in community detection," *Proc. Nat. Academy Sci. United States America*, vol. 104, no. 1, pp. 36–41, 2007.

[53] S. Wu, A. Joseph, A. Hammonds, et al., "Stability-driven nonnegative matrix factorization to interpret spatial gene expression and build local gene networks," *Proc. Nat. Academy Sci. United States America*, vol. 113, no. 16, pp. 4290–4295, 2016.

[54] C. J. Lin, "Projected gradient methods for non-negative matrix factorization," *Neural Comput.*, vol. 19, no. 10, pp. 2756–2779, 2007.

[55] D. Mavroeidis, "Accelerating spectral clustering with partial supervision," *Data Mining Knowl. Discovery*, vol. 21, no. 2, pp. 241–258, 2010.

[56] O. Vanunu, O. Magger, E. Ruppin, et al., "Associating genes and protein complexes with disease via network propagation," *PLoS Comput. Biol.*, vol. 6, no. 1, 2010, Art. no. e1000641.

[57] D. Zhou, O. Bousquet, T. N. Lal, et al., "Learning with local and global consistency," *Adv. Neural Inf. Process. Syst.*, vol. 16, no. 4, pp. 321–328, 2004.

[58] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, et al., "Gene ontology: Tool for the unification of biology," *Nature Genetics*, vol. 25, no. 1, pp. 25–29, 2000.

[59] A. D. King, N. Prulj, and I. Jurisica, "Protein complex prediction via cost-based clustering," *Bioinf.*, vol. 20, no. 17, pp. 3013–3020, 2004.

[60] Y. Benjamini and Y. Hochberg, "Controlling the false discovery rate: A practical and powerful approach to multiple testing," *J. Roy. Statistical Society B Methodological*, vol. 57, no. 57, pp. 289–300, 1995.

[61] N. V. Jordan, A. Bardia, et al., "HER2 expression identifies dynamic functional states within circulating breast cancer cells," *Nature*, vol. 537, pp. 102–106, 2016.

[62] J. Leskovec, L. Adamic, and B. Adamic, "The dynamics of viral marketing," *ACM Trans. Web*, vol. 1, 2007, Art. no. 1.

[63] X. Shen, W. Liu, I. Tsang et al., "Compressed K-means for large-scale clustering," in *Proc. 31th AAAI Conf. Artif. Intell.*, 2017, pp. 2527–2533.

[64] W. Liu, X. Shen, and I. Tsang, "Sparse embedded k-means clustering," in *Proc. 31th Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 3321–3329.

Xiaoke Ma received the PhD degree in computer science from Xidian University, in 2012. He is an associate professor with the School of Computer Science and Technology, Xidian University (P.R. China). His research interests include machine learning, data mining, and bioinformatics. He publishes more than 40 papers in the peer-reviewed international journals, such as the *IEEE Transactions on Knowledge and Data Engineering*, the *IEEE Transactions on Cybnetics*, *Pattern Recognition*, *Bioinformatics*, *PLoS Computational Biology*, *Cell Stem Cell*, *Nuclear Acids Research*, and the *IEEE/ACM Transactions on Computational Biology and Bioinformatics*.

Di Dong received the PhD degree in computer science from the Chinese Academy of Sciences. At present, he is an assistant professor in the Institute of Automation, Chinese Academy of Sciences. His research interests include data mining and medical image processing. He has published 10 papers in international journals.

Quan Wang received the BS, MS, and PhD degrees in computer science and technology from Xidian University, Xian, China. He is now a full professor with Xidian University. His current research interests include input and output technologies and systems, image processing, and data mining.