A Survey of Community Detection in Complex Networks Using Nonnegative Matrix Factorization

Chaobo He, Xiang Fei, Qiwei Cheng, Hanchao Li, Zeng Hu, and Yong Tang

***Abstract*—Community detection is one of the popular research topics in the field of complex networks analysis. It aims to identify communities, represented as cohesive subgroups or clusters, where nodes in the same community link to each other more densely than others outside. Due to the interpretability, simplicity, flexibility, and generality, nonnegative matrix factorization (NMF) has become a very ideal model for community detection and lots of related methods have been presented. To facilitate research on NMF-based community detection, in this article, we make a comprehensive review on NMF-based methods for community detection, especially the state-of-the-art methods presented in high prestige journals or conferences. First, we introduce the basic principles of NMF and explain why NMF can detect communities and design a general framework of NMF-based community detection. Second, according to the applicable network types, we propose a taxonomy to divide the existing NMF-based methods for community detection into six categories, namely, topology networks, signed networks, attributed networks, multilayer networks, dynamic networks, and large-scale networks. We deeply analyze representative methods in every category. Finally, we summarize the common problems faced by all methods and potential solutions and propose four promising research directions. We believe that this survey can fully demonstrate the versatility of NMF-based community detection and serve as a useful guideline for researchers in related fields.**

***Index Terms*—Attributed networks, community detection, complex networks, dynamic networks, large-scale networks, multilayer networks, nonnegative matrix factorization (NMF), signed networks, topology networks.**

Nomenclature

G	Given complex network.
V	Nodes set.
v_i	ith node.
E	Edges set.
e_{ij}	Edge from v_i to v_j.
n	Number of nodes.
t	Number of iterations.
k	Number of communities.
C	Communities set.
C_i	ith community.
$\mathbf{X}$	Feature matrix.
$\mathbf{A}$	Adjacency matrix.
$\mathbf{I}$	Identity matrix.
$\mathbf{1}$	Matrix whose elements are all 1.
$\mathbf{H}$	Community indicator matrix.
$\mathbb{R}_+$	Nonnegative real number set.
$\circ$	Elementwise multiplication operator.

Manuscript received April 30, 2021; revised August 19, 2021; accepted September 14, 2021. Date of publication October 5, 2021; date of current version April 1, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 62077045, Grant U1811263, and Grant 61772211; in part by the Humanity and Social Science Youth Foundation of Ministry of Education of China under Grant 19YJCZH049; and in part by the Natural Science Foundation of Guangdong Province of China under Grant 2019A1515011292. *(Corresponding author: Yong Tang.)*

Chaobo He is with the School of Information Science and Technology, Zhongkai University of Agriculture and Engineering, Guangzhou 510225, China, and also with the School of Computer Science, South China Normal University, Guangzhou 510631, China (e-mail: hechaobo@foxmail.com).

Xiang Fei and Hanchao Li are with the Department of Computing, Coventry University, Coventry CV1 5FB, U.K. (e-mail: aa5861@coventry.ac.uk; lih30@uni.coventry.ac.uk).

Qiwei Cheng and Yong Tang are with the School of Computer Science, South China Normal University, Guangzhou 510631, China (e-mail: 2019022653@m.scnu.edu.cn; ytang@m.scnu.edu.cn).

Zeng Hu is with the School of Information Science and Technology, Zhongkai University of Agriculture and Engineering, Guangzhou 510225, China (e-mail: huzeng@zhku.edu.cn).

Digital Object Identifier 10.1109/TCSS.2021.3114419

I. Introduction

Complex networks are powerful tools used to model various complex systems, including social systems, information systems, and ecosystems. They are very common in real world, such as social networks, coauthorship networks, communication networks, and protein–protein interaction (PPI) networks. Because complex networks often contain rich information, they have drawn considerable attention from researchers. Many research topics of complex networks analysis are constantly emerging. Among them, community detection is very attractive. It is generally believed that a community (also referred to as a partition, a subgraph, a module, or a cluster) is a group of cohesive nodes, within which nodes are connected more densely than those outside [1], [2]. Detecting community effectively is not only very useful for understanding the structures and functions of complex networks but also of great value in practical applications. For example, it can be used to find research teams in coauthorship networks, protein complexes in PPI networks, and groups of similar users in online social networks. Besides, community detection is also an interdisciplinary research topic, which mainly involves sociology, physics, mathematics, and computer science. These characteristics make community detection in complex networks a popular topic of great research value.

Recently, various methods for community detection have been proposed, such as spectral clustering-based methods [3], [4], stochastic block model-based methods [5], label propagation-based methods [6], game theory-based methods [7], [8], and deep learning-based methods [9]. It is worth noting that nonnegative matrix factorization (NMF)-based methods have also received a lot of attention. Many related works have been constantly presented in influential international conferences (e.g., AAAI, IJCAI, KDD, ICDM, CIKM, WSDM, NIPS, and ICPR) and high-quality peer-reviewed journals (e.g., PNAS, TPAMI, TKDE, and TNNLS) in the area of artificial intelligence, machine learning, and data mining. Comparing with other models used for community detection, NMF has fully demonstrated some unique advantages as follows.

1) *Higher Interpretability to Community Detection Results:* Given a complex network, we can represent it as a nonnegative feature matrix (e.g., the adjacency matrix). Through NMF, we can factorize this feature matrix to obtain a node-community indicator matrix. Due to the nonnegative constraints, every element in this matrix can be naturally treated as the strength of the corresponding node belonging to the corresponding community. This makes the community detection results more interpretable.
2) *More Simple and Effective to Detect Overlapping Communities:* For overlapping communities, which are very commonly occurred in real-world complex networks, a node is allowed to belong to multiple communities. In response, NMF has an inherent soft clustering ability that can learn the community membership distribution of every node. In the postprocessing step, we only need to use a predetermined strength threshold to decide which communities a node should be assigned into, and hence, overlapping communities can be easily extracted.
3) *More Flexible to Incorporate Prior Knowledge:* Lots of existing works have shown that effectively using prior knowledge (e.g., node's community labels, must-link, and cannot-link constraints) can improve the performance of community detection. In view of this, NMF provides two strategies to incorporate this prior knowledge. One is to transform prior knowledge to one part of the feature matrix used for NMF. Another is to transform prior knowledge to the regularized constraint term used for guiding the learning process of NMF-based community detection model. More importantly, these two ways both set weights to balance the contribution of prior knowledge.
4) *More General to Detect Communities in Various Complex Networks:* Real-world complex networks have many types, such as directed/undirected networks, signed networks, attributed networks, multilayer networks, and dynamic networks. NMF and its variants (it usually does not require too many extensions) can effectively deal with the problem of community detection in any kind of these complex networks. This is often not possible for other community detection models. It could be argued that NMF is very versatile in terms of community detection.

Because of these beneficial characteristics, NMF has become a very ideal model for community detection in complex networks. Actually, it is also being used more and more widely, and getting more and more attention. In this survey, we give a comprehensive review of the state-of-the-art NMF-based community detection methods. From this survey, we hope to provide a useful guideline for researchers in related fields to understand: 1) the basic theories of NMF-based community detection; 2) the methods' taxonomy according to the types of networks and the characteristics of different types of methods; and 3) the common problems and their solutions, and the future research directions. To the best of our knowledge, this is the first work to provide a comprehensive review of NMF-based community detection methods in English. Specifically, our main contributions include three aspects.

1) We reveal the principles that justify why NMF can be used to identify community structure in complex networks and design a general framework for NMF-based community detection.
2) We propose a new categorization of the existing NMF-based community detection methods according to the types of networks to which they are applicable to and provide a detailed and in-depth introduction to the representative methods.
3) We summarize the common problems faced by all NMF-based community detection methods and the corresponding solutions, along with suggestions of promising opportunities for future works.

The rest of this survey is organized as follows. In Section II, we introduce notations and preliminaries required to understand the problem and the models discussed in the following. Section III gives an in-depth analysis to the basic theories of NMF-based community detection. Section IV proposes taxonomy to categorize the existing NMF-based community methods. In Section V, we summarize the common problems encountered by all methods and their potential solutions. Section VI discusses future research directions, and finally, we conclude this article in Section VII.

II. Notations and Preliminaries

Throughout this article, we denote matrices by bold uppercase letters. For a given matrix $\mathbf{X}$, its ith row vector, jth column vector, $(i,\ j)$th element, trace, transpose, and Frobenius norm are denoted by $\mathbf{X}_{i.}$, $\mathbf{X}_{.j}$, $\mathbf{X}_{ij}$, $tr(\mathbf{X})$, $\mathbf{X}^T$, and $\|\mathbf{X}\|_F$, respectively. For ease of presentation, we summarize a list of frequently used notations in the Nomenclature.

Definition 1 (Complex Network): Following the graph theory, a given complex network can be denoted as a graph $G = (V, E)$, where $V = \{v_1, v_2, \ldots, v_n\}$ and $E = \{e_{ij}|v_i \in V \wedge v_j \in V\}$. In general, G can be described by an adjacency matrix $\mathbf{A} = [\mathbf{A}_{ij}]^{n\times n}$, where $\mathbf{A}_{ij}$ characterizes the relationship between v_i and v_j. For an unweighted G, we have $\mathbf{A}_{ij} = 1$ if $e_{ij} \in E$ and $\mathbf{A}_{ij} = 0$ otherwise. If G is weighted, then $\mathbf{A}$ is real-valued. Besides, $\mathbf{A}$ is symmetric if G is undirected; otherwise, it is not necessarily symmetric. Note that we will redefine $\mathbf{A}$ for some special complex networks, such as signed networks which will be introduced in Section IV-B.

Definition 2 (Community): Essentially, communities are the subgraphs of G, where nodes have dense internal connections and sparse external connections. Supposing that G comprises k communities, we denote the communities set as

$C = \{C_i | C_i \neq \emptyset, 1 \leq i \leq k\}$. Due to the possibility of overlapping, the intersection of C_i and C_j $(i \neq j)$ may not be empty.

Based on the definitions above, the goal of community detection is to identify k communities in G using a specific model, such as NMF focused in this article.

III. NMF and Community Detection

In this section, we first give a brief introduction to NMF and then explain why NMF can detect communities in complex networks. Finally, we design a general framework for NMF-based community detection.

A. NMF

NMF formally proposed by Lee and Seung [10] is a classical low-rank matrix factorization model. It is specially applicable for analyzing the matrices whose elements are all nonnegative. Mathematically, given a nonnegative feature matrix $\mathbf{X} = [\mathbf{X}_{.1}, \mathbf{X}_{.2}, \ldots, \mathbf{X}_{.n}] \in \mathbb{R}_+^{m \times n}$ composed of n m-dimension data vectors, and the desired reduced dimension d $(d \ll \min(m, n))$, NMF aims to find two nonnegative matrices $\mathbf{W} = [\mathbf{W}_{ip}]^{m \times d} \in \mathbb{R}_+^{m \times d}$ and $\mathbf{H} = [\mathbf{H}_{jp}]^{n \times d} \in \mathbb{R}_+^{n \times d}$, which can well approximate to the original matrix $\mathbf{X}$ in the form of their product

$$\mathbf{X} \approx \mathbf{W}\mathbf{H}^T \tag{1}$$

where $\mathbf{W}$ and $\mathbf{H}$ are, respectively, called the basis matrix and the coefficient matrix. Due to the nonnegativity constraints on $\mathbf{W}$ and $\mathbf{H}$, every data sample $\mathbf{X}_{.j} \in \mathbf{X}$ can be represented as an additive linear combination of the basis vectors $\mathbf{W}_{.p} \in \mathbf{W}$ $(1 \leq p \leq d)$, i.e., $\mathbf{X}_{.j} \approx \sum_{p=1}^{d} \mathbf{W}_{.p}\mathbf{H}_{jp}$. This feature naturally conforms to the intuitive human cognition of "combining parts to form a whole," which makes NMF have high physical interpretability. Meanwhile, this also indicates that NMF is a linear model. Recently, there have been some works that tried to turn NMF into the nonlinear model, such as kernel NMF [11] and nonlinear projective NMF [12].

To obtain $\mathbf{W}$ and $\mathbf{H}$ in (1), we can solve the objective function that minimizes the approximation error of (1). One objective function is the square of the Frobenius norm of the difference between $\mathbf{X}$ and $\mathbf{W}\mathbf{H}^T$

$$\min \mathcal{L}(\mathbf{W}, \mathbf{H}) = \left\| \mathbf{X} - \mathbf{W}\mathbf{H}^T \right\|_F^2, \quad \text{s.t.} \quad \mathbf{W} \geq 0, \mathbf{H} \geq 0. \tag{2}$$

$\mathcal{L}(\mathbf{W}, \mathbf{H})$ is not convex to $\mathbf{W}$ and $\mathbf{H}$ together, so it is unrealistic to expect an algorithm to find the global minimum of $\mathcal{L}(\mathbf{W}, \mathbf{H})$. Lee and Seung [13] developed an iterative update algorithm shown as follows to solve $\min \mathcal{L}(\mathbf{W}, \mathbf{H})$ optimally. Meanwhile, they proved that this algorithm can well guarantee the convergence of $\mathcal{L}(\mathbf{W}, \mathbf{H})$

$$\mathbf{W}_{ip} = \mathbf{W}_{ip} \frac{(\mathbf{X}\mathbf{H})_{ip}}{\left(\mathbf{W}\mathbf{H}^T\mathbf{H}\right)_{ip}}, \quad \mathbf{H}_{jp} = \mathbf{H}_{jp} \frac{\left(\mathbf{X}^T\mathbf{W}\right)_{jp}}{\left(\mathbf{H}\mathbf{W}^T\mathbf{W}\right)_{jp}}. \tag{3}$$

Although there are some other types of objective functions, which also can quantify the approximation error between $\mathbf{X}$ and $\mathbf{W}\mathbf{H}^T$, such as Kullback–Leibler (KL) divergence, Bregman divergence, and I-divergence introduced in [14], the square of the Frobenius norm above is the most widely used due to its simplicity and effectiveness. Besides, it should be pointed out that most existing solution algorithms of NMF and its variants follow or can be transformed to the iterative update framework shown in (3).

B. Why NMF Can Detect Communities?

Compared with other types of matrix factorization techniques (e.g., LU factorization, Cholesky factorization, QR factorization, and singular value decomposition (SVD) factorization summarized in [15]), NMF is more widely used in image representation [16], dimensionality reduction [17], and recommender system [18] due to its high interpretability. Especially, NMF is more suitable for the task of community detection. This is because it has two unique capabilities. One is the potential clustering capability possessed by NMF. In [19], NMF and its extensions are proved to have equivalent relationships with some classical clustering models. For example, if we let $\mathbf{H}\mathbf{H}^T = \mathbf{I}$ (i.e., imposing orthogonal constraints on $\mathbf{H}$), then NMF shown in (2) is equivalent to k-means clustering model. In this case, $\mathbf{W}$ and $\mathbf{H}$ are called cluster centroids and cluster indicator matrices, respectively. The reduced dimension d is equal to the number of clusters. If the square $\mathbf{X}$ is symmetric, NMF can be further transformed to the symmetric decomposition form as $\mathbf{X} \approx \mathbf{H}\mathbf{H}^T$, which is equivalent to spectral clustering model. Essentially, community detection is a clustering problem, whose clustering objects are nodes in complex networks. Both k-means and spectral clustering models show their effectiveness in dealing with the problem of node clustering [3], [4]. Therefore, NMF can be naturally used to detect communities. In fact, most of the existing NMF-based methods for community detection obtain better performances by improving the clustering ability of NMF.

The other aspect is the generative capability of NMF that can give a good interpretation to community structure [20]. When NMF is used to detect communities, the corresponding adjacency matrix $\mathbf{A}$ is often selected as the feature matrix used for factorization, i.e., $\mathbf{A} \approx \mathbf{W}\mathbf{H}^T$. In this context, $\mathbf{W}$ and $\mathbf{H}$, respectively, denote the community feature matrix and community indicator matrix, and the reduced dimension d is the number of communities k. $\forall \mathbf{H}_{jp} \in \mathbf{H}$ represents the strength of υ_j belonging to the pth community. The product of $\mathbf{W}_{ip} \in \mathbf{W}$ and $\mathbf{H}_{jp} \in \mathbf{H}$ can be treated as the expected interactions between υ_i and υ_j, which are deduced by their mutual participation in the pth community. Based on this, by summing over all the k communities, we can obtain the total expected interactions between υ_i and υ_j as $\sum_{p=1}^{k} \mathbf{W}_{ip}\mathbf{H}_{jp}$. This implies that if υ_i and υ_j share more communities, they have more interactions, which will result in higher probability that they will be connected. Namely, $\mathbf{A}_{ij} \approx \sum_{p=1}^{k} \mathbf{W}_{ip}\mathbf{H}_{jp}$, which is consistent with the NMF-based community detection model above and can well explain why nodes in the same communities are densely connected.

C. General Framework for NMF-Based Community Detection

Although existing NMF-based methods for community detection have different model characteristics, they all consist

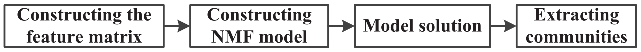

Fig. 1. Workflow for NMF-based community detection.

of four key processing stages, i.e., constructing the feature matrix, constructing NMF-based community detection model, model solution, and extracting communities. The corresponding workflow is shown in Fig. 1.

As the first stage, constructing the feature matrix is mainly responsible for extracting features from G and representing them as the feature matrix $\mathbf{X}$. Undeniably, more accurate feature matrix can help to obtain better performance of community detection. In the second and the third stages, the specific NMF-based community detection model is designed to factorize $\mathbf{X}$ to obtain the community indicator matrix $\mathbf{H}$. How to obtain more accurate $\mathbf{H}$ and improve the algorithm efficiency are the focuses of these two stages. In the stage of extracting communities, communities can be easily inferred based on $\mathbf{H}$ no matter whether they are overlapping or not. Specifically, for the given υ_j, in the case of nonoverlapping community detection, we only need to assign υ_j into the pth community satisfying the requirement that $\mathbf{H}_{jp}$ is the maximum element in $\mathbf{H}_{j\cdot}$. In the case of overlapping community detection, we first need to set a threshold ϕ, and then, υ_j is assigned into the pth community as long as $\mathbf{H}_{jp} > \phi$. Through this way, υ_j is possible to be assigned into multiple communities.

According to the aforementioned descriptions, in Algorithm 1, we design a general framework for NMF-based community detection, which is composed of the above four common stages. To present this framework more clearly, we use a toy example shown in Fig. 2 to illustrate how it works. It can be said that almost every NMF-based method for community detection all can be simplified into this framework. For these two input parameters k and ϕ, some automatic optimal setting schemes have been proposed, which are, respectively, discussed in Session V-C and Session V-D.

Algorithm 1 NMF-Based Community Detection Framework

Input: $G = (V, E), k, \phi$;
Output: Communities set $C = \{C_1, C_2, \ldots, C_k\}$;
Constructing the feature matrix $\mathbf{X}$;
Constructing NMF-based community detection model like: $\min \mathcal{L}(\mathbf{W}, \mathbf{H})$;
$\mathbf{H} \leftarrow$ Solving $\min \mathcal{L}(\mathbf{H})$;
for $\upsilon_j \in V$ **do**
  if *non-overlapping* **then**
    $q = \operatorname{argmax}_p \mathbf{H}_{jp}$;
    $C_q = C_q \bigcup \{\upsilon_j\}$;
  if *overlapping* **then**
    for $\forall \mathbf{H}_{jp} \in \mathbf{H}_{j\cdot}$ **do**
      if $\mathbf{H}_{jp} > \phi$ then $C_p = C_p \bigcup \upsilon_j$;
return C;

IV. NMF-Based Community Detection Methods for Various Complex Networks

To demonstrate the versatility of NMF and its variants in dealing with the problem of community detection, we present a taxonomy of NMF-based community detection methods, as shown in Table I. We divide the existing methods into six categories according to the types of networks that they are specially applicable to, including topology networks, signed networks, attributed networks, multilayer networks, dynamic networks, and large-scale networks. In the following, we will overview the representative methods in each category.

A. Topology Networks

Here, we call complex networks that only contain topology structure information (i.e., links information) as topology networks. These networks can be divided into directed networks and undirected networks according to whether the links are directed. For directed or undirected networks, the conventional NMF model, $\mathbf{X} \approx \mathbf{W}\mathbf{H}^T$, can be directly used to detect communities by replacing $\mathbf{X}$ with $\mathbf{A}$. However, it cannot model the interactions among communities, which is helpful to determine whether communities are overlapping. In view of this, Wang *et al.* [21] proposed a nonnegative matrix trifactorization model (NMTF)

$$\min \mathcal{L}(\mathbf{H}, \mathbf{S}) = \left\|\mathbf{A} - \mathbf{H}\mathbf{S}\mathbf{H}^T\right\|_F^2, \quad \text{s.t.} \quad \mathbf{H} \geq 0, \mathbf{S} \geq 0 \tag{4}$$

where $\mathbf{S} = [\mathbf{S}_{ij}]^{k\times k} \in \mathbb{R}_+^{k\times k}$ denotes the interactions among all communities. Zhang and Yeung [22] further extended this model and proposed a bounded NMTF (BNMTF) model, whose performance is better than NMTF.

In NMTF, if G is undirected, then $\mathbf{A}$ is symmetric, and it can be deduced that $\mathbf{S}$ is also symmetric. Besides, $\mathbf{S}$ is semipositive definite because $\mathbf{H}\mathbf{S}\mathbf{H}^T$ is always greater than or equal to 0 under the nonnegative constraints on $\mathbf{H}$ and $\mathbf{S}$. These features can lead to an interesting transformation: let $\mathbf{S} = \mathbf{S}^{(1/2)}\mathbf{S}^{(1/2)}$, then $\mathbf{H}\mathbf{S}\mathbf{H}^T = (\mathbf{H}\mathbf{S}^{(1/2)})(\mathbf{H}\mathbf{S}^{(1/2)})^T$, which means that $\mathbf{S}$ can be absorbed into $\mathbf{H}$, i.e., $\mathbf{H} = \mathbf{H}\mathbf{S}^{(1/2)}$. As a result, we can obtain a simplified form of NMTF

$$\min \mathcal{L}(\mathbf{H}) = \left\|\mathbf{A} - \mathbf{H}\mathbf{H}^T\right\|_F^2, \quad \text{s.t.} \quad \mathbf{H} \geq 0 \tag{5}$$

where $\mathbf{H}$ still denotes the community indicator matrix. This model is called symmetric NMF (SNMF) and has only one factor matrix $\mathbf{H}$, which makes it more efficient than NMTF.

Although directed networks are the most common type of networks in real world, we find that most of the NMF-based community detection methods prefer to model networks as undirected networks. Moreover, many researchers like to devise various SNMF variants to improve the performance of community detection in undirected networks, among which variants based on graph regularized SNMF (GRSNMF) are the most common ones. The general formulation of GRSNMF is as follows:

$$\min \mathcal{L}(\mathbf{H}) = \left\|\mathbf{A} - \mathbf{H}\mathbf{H}^T\right\|_F^2 + \lambda tr\left(\mathbf{H}^T\mathbf{L}\mathbf{H}\right), \quad \text{s.t.} \quad \mathbf{H} \geq 0 \tag{6}$$

where $\mathbf{L} \in \mathbb{R}^{n\times n}$ is the Laplacian matrix of a certain constraint information matrix (e.g., node similarity matrix or geometric

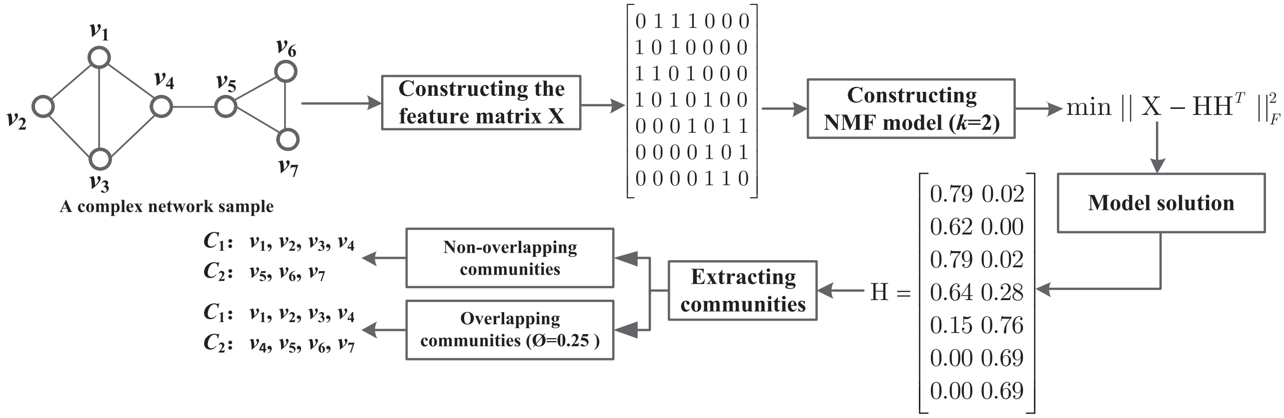

Fig. 2. Illustration example of NMF-based community detection. Note that **H** here is not the unique solution because the results of multiplying **H** by any square semipositive definite matrix are all possible solutions. This can refer to the analysis of NMTF model introduced in Section IV-A.

TABLE I
TAXONOMY AND REPRESENTATIVE METHODS

Category	Representative methods
Topology networks	NMTF [21], BNMTF [22], PCSNMF [23], PSSNMF [24], HPNMF [25], HNMF [26], A^{2}NMF [27], PNMF [28]
Signed networks	JNMF [31], SGNMF [32], MCNMF [33], ReS-NMF [36], BRSNMF [37], SPOCD [38]
Attributed networks	FSL [40], JWNMF [41], NMTFR [42], CFOND [43], SCI [44], ASCD [45], DII [46], RSECD [47]
Multi-layer networks	WSSNMTF [50], NF-CCE [51], MTRD [53], LJ-SNMF [54], S2-jNMF [55]
Dynamic networks	sE-NMF [57], GrENMF [58], Cr-ENMF [59], ECGNMF [60], DGR-SNMF [61], DBNMF [62], C^3 [66], Chimera [70]
Large-scale networks	BIGCLAM [73], HierSymNMF2 [75], cyclicCDSymNMF [77], OGNMF [79], DRNMFSR [80], TCB [81]

structure matrix), $tr(\mathbf{H}^T\mathbf{LH})$ is called the graph regularized term, and λ is the regularized parameter. By minimizing $\mathcal{L}(\mathbf{H})$, $tr(\mathbf{H}^T\mathbf{LH})$ as the constraint term is able to guide GRSNMF to learn more accurate community indicator matrix $\mathbf{H}$.

Generally, methods based on GRSNMF improve the performance through integrating extra information with the link information, so what kinds of extra information can be integrated is their focuses. Recently, two kinds of information are often utilized. One is prior knowledge, also known as semisupervised information, such as ground-truth community labels, node must-link, and cannot-link constraints. For example, PCSNMF [23] and PSSNMF [24] both first transform the prior information to the constraint information matrix and then use GRSNMF to achieve good performance. The other type of extra information is the intrinsic constraint information extracted from the link information itself, such as node homogeneity information used in HPNMF [25] and HNMF [26], node affinity information used in A^{2}NMF [27], and link preference information used in PNMF [28]. For methods based on GRSNMF, integrating the intrinsic constraint information is more operable than integrating prior knowledge. After all, prior knowledge is often not available due to the difficulty in acquiring it in many real-world complex networks, especially in large-scale complex networks.

B. Signed Networks

Complex networks containing explicit positive or negative edges are called signed networks [29]. Positive edges in signed networks denote the positive relationships (e.g., "friend" and "trust"). Conversely, negative edges denote the negative relationships (e.g., "enemy" and "distrust"). Signed networks are very common in real world, such as Bitcoin,[1] Slashdot[2] signed social networks, and international relationship network. For a given signed network, due to the existence of edge signs its adjacency matrix, $\mathbf{A} = [\mathbf{A}_{ij}]^{n\times n}$ is redefined as follows:

$$\mathbf{A}_{ij} = \begin{cases} 1, & \text{if the edge from } v_i \text{ to } v_j \text{ is positive} \\ -1, & \text{if the edge from } v_i \text{ to } v_j \text{ is negative} \\ 0, & \text{otherwise.} \end{cases} \quad (7)$$

Being different from communities in unsigned networks, communities in signed networks not only require dense intracommunity and sparse intercommunity links but also that most positive links should lie within communities and most negative links should lie between communities. This requirement makes community detection in signed networks more challenging than community detection in unsigned networks. Recently, some NMF-based methods for community detection in signed networks have been proposed. These methods can be roughly categorized into two types: methods based on joint NMF and methods based on Semi-NMF which is an NMF variant proposed by Ding *et al.* [30].

The basic idea of joint NMF-based methods is that they first divide $\mathbf{A}$ into two parts: $\mathbf{A}^+ \in \mathbb{R}_+^{n\times n}$ and $\mathbf{A}^- \in \mathbb{R}_+^{n\times n}$, which, respectively, denote positive edges and negative edges, and then use NMF to jointly factor $\mathbf{A}^+$ and $\mathbf{A}^-$ to obtain

[1] https://bitcoin.org/en/
[2] https://slashdot.org/

a consensus community indicator matrix $\mathbf{H}$. Note that every element in $\mathbf{A}^+$ and $\mathbf{A}^-$ uses 1 or 0 to represent the existence of the corresponding edge. Following this idea, Yan *et al.* [31] presented a method JNMF that unites two weighted NMTF models

$$\begin{aligned} \min \mathcal{L}(\mathbf{H}, \mathbf{S}_1, \mathbf{S}_2) = & \left\|\mathbf{B} \circ \left(\mathbf{A}^+ - \mathbf{H}\mathbf{S}_1\mathbf{H}^T\right)\right\|_F^2 \\ & + \left\|\mathbf{B} \circ \left(\mathbf{A}^- - \mathbf{H}\mathbf{S}_2\mathbf{H}^T\right)\right\|_F^2 \\ & \text{s.t.} \quad \mathbf{H} \geq 0, \mathbf{S}_1 \geq 0, \mathbf{S}_2 \geq 0 \end{aligned} \tag{8}$$

where $\mathbf{B}$ is the weight matrix containing priorities of links to community structures and $\mathbf{S}_1$ and $\mathbf{S}_2$ both denote community interaction matrices. Similar to JNMF, methods SGNMF [32] and MCNMF [33] both jointly factor $\mathbf{A}^+$ and $\mathbf{A}^-$ to obtain the consensus $\mathbf{H}$. However, they still have a little difference that they introduce graph regularized items into the joint NMF model. This enables them to integrate more information to improve the performance.

Unlike methods based on joint NMF, methods based on Semi-NMF can directly factor $\mathbf{A}$ to detect communities without separating it into $\mathbf{A}^+$ and $\mathbf{A}^-$ in advance. This is because Semi-NMF removes the nonnegative constraint on the feature matrix used for factorization and meanwhile still retains good clustering ability [34], [35]. Based on the Semi-NMF model, Li *et al.* [36] proposed a method called ReS-NMF. It introduces a graph regularization to distribute the pair of nodes that are connected with negative links into different communities. Shi *et al.* [37] also proposed a method BRSNMF, which is based on the regularized Semi-NMF, but BRSNMF introduces two graph regularized items

$$\begin{aligned} \min \mathcal{L}(\mathbf{W}, \mathbf{H}) = & \left\|\mathbf{A} - \mathbf{W}\mathbf{H}^T\right\|_F^2 + \alpha tr\left(\mathbf{H}\mathbf{1}\mathbf{H}^T\right) \\ & - \beta tr\left(\mathbf{H}^T(\sigma\mathbf{I} - \eta\mathbf{L})\mathbf{H}\right) \\ & \text{s.t.} \quad \mathbf{H} \geq 0 \end{aligned} \tag{9}$$

where the first regularized item is used to control the sparsity of $\mathbf{H}$, the second one is used to encode the balance structure information of signed networks, $\mathbf{L}$ is the Laplacian matrix of $\mathbf{A}$, $\sigma, \eta > 0$ control the sizes of communities, and α and β are both regularized parameters. Note that methods based on Semi-NMF all only need to impose a nonnegative constraint on $\mathbf{H}$. Through experiments, ReS-NMF and BRSNMF both fully demonstrate the feasibility and effectiveness of detecting communities in signed networks using Semi-NMF.

In [38], we also proposed a semi-NMF-based method named SPOCD. Unlike ReS-NMF and BRSNMF, we specially design a node similarity measure to extract node similarity information from signed networks and use this new information matrix instead of $\mathbf{A}$ as the feature matrix. Considering that communities in signed networks are possible overlapping, we further introduce a discrete optimization strategy to obtain the binary $\mathbf{H}$, which makes overlapping communities detection more accurate. Experimental results show that SPOCD performs better than ReS-NMF and BRSNMF.

C. Attributed Networks

Many complex networks not only have link information but also have attribute information. For example, users in online social networks are associated with demographic attributes (e.g., age, gender, and occupation). Besides, some other feature attributes can be extracted from their generated content, including user profiles and posts. Complex networks such as these are called attributed networks. Recently, it is generally believed that link information alone is not sufficient to detect high-quality communities because the link information in real-world complex networks may be noisy or incomplete. In this situation, the attribute information is often considered as a good supplement to the link information due to its availability.

How to effectively fuse link information and attribute information is the focus of the methods for community detection in attributed networks [39]. In this aspect, many NMF-based methods demonstrate their superiorities comparing with other types of methods. In general, most of them use variants of the joint NMF to fuse these two types of information. One of the common variants is the consensus factorization model adopted in methods FSL [40], JWNMF [41], NMTFR [42], and CFOND [43]. In this model, the link information is denoted as the adjacency matrix $\mathbf{A}$, and the attribute information is denoted as a node attribute matrix $\mathbf{Y} = [\mathbf{Y}_{ij}]^{m\times n} \in \mathbb{R}_+^{m\times n}$, where m is the length of the attribute set $(a_1, a_2, \ldots, a_m)$ and $\mathbf{Y}_{ij}$ can be simply defined as follows:

$$\mathbf{Y}_{ij} = \begin{cases} 1, & \text{if the } a_i \text{ is the attribute of } v_j \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

After obtaining $\mathbf{A}$ and $\mathbf{Y}$, the consensus factorization model, respectively, factorizes $\mathbf{A}$ and $\mathbf{Y}$ using NMF and meanwhile forcing them to have a common factor $\mathbf{H} \in \mathbb{R}_+^{n\times k}$ used to represent the community memberships. If $\mathbf{A}$ is symmetric, this consensus factorization model can be simply formulated as the following optimization problem:

$$\begin{aligned} \min \mathcal{L}(\mathbf{W}, \mathbf{H}) = & \left\|\mathbf{A} - \mathbf{H}\mathbf{H}^T\right\|_F^2 + \alpha\left\|\mathbf{Y} - \mathbf{W}\mathbf{H}^T\right\|_F^2 \\ & \text{s.t.} \quad \mathbf{W} \geq 0, \mathbf{H} \geq 0 \end{aligned} \tag{11}$$

where α is used to balance the contribution of the attribute information and $\mathbf{W} \in \mathbb{R}_+^{m\times k}$ is the community attribute matrix used to infer the most relevant attributes for each community.

Another common variant of joint NMF for community detection in the attributed network is the chain factorization model utilized in methods SCI [44], ASCD [45], and DII [46]. Unlike the consensus factorization model, the chain factorization model does not factorize $\mathbf{A}$ and $\mathbf{Y}$ simultaneously. It factors $\mathbf{A}$ to obtain $\mathbf{H}$ and meanwhile factors $\mathbf{H}$ as the product of $\mathbf{Y}$ and the community attribute matrix $\mathbf{W}$. The corresponding united model is

$$\begin{aligned} \min \mathcal{L}(\mathbf{W}, \mathbf{H}) = & \left\|\mathbf{A} - \mathbf{H}\mathbf{H}^T\right\|_F^2 + \alpha\left\|\mathbf{H} - \mathbf{Y}^T\mathbf{W}\right\|_F^2 \\ & \text{s.t.} \quad \mathbf{W} \geq 0, \mathbf{H} \geq 0. \end{aligned} \tag{12}$$

Equation (12) can be interpreted from the perspective of generative model. Its first part models the process of generating links between nodes: if two nodes have similar community memberships, they have a high possibility to be linked, and its second part models the process of generating communities: if the attributes of a node are highly similar to those of a community, the node may have a high possibility to be in this community.

Essentially, these two models aforementioned both assume that the attribute information shares the same cluster (i.e., community) structure with the link information. Hence, if complex networks have adequate and reliable attribute information, these two models can be expected to obtain good performance, but if the attribute information is poor, they may perform worse than using the link information alone. In view of this, Jin *et al.* [47] designed a robust joint NMF model named RSECD to fuse the link information and the attribute information more intelligently. Formally, this model is denoted as the following objective function:

$$\begin{aligned}\min \mathcal{L}(\mathbf{W},\mathbf{H},\mathbf{U},\mathbf{V}) = {} & \left\|\mathbf{A}-\mathbf{H}\mathbf{H}^T\right\|_F^2 + \alpha\left\|\mathbf{Y}-\mathbf{W}\mathbf{V}^T\right\|_F^2 \\ & + \|\mathbf{H}\mathbf{U}-\mathbf{V}\|_F^2 + \|\mathbf{I}-\mathbf{U}\|_F^2 \\ & + \left\|\mathbf{U}\mathbf{1}_k^T-\mathbf{1}_k^T\right\|_F^2 \\ & \text{s.t.}\quad \mathbf{W}\geq 0, \mathbf{H}\geq 0, \mathbf{U}\geq 0, \mathbf{V}\geq 0\end{aligned} \tag{13}$$

where $\mathbf{H} \in \mathbb{R}_+^{n\times k}$ and $\mathbf{V} \in \mathbb{R}_+^{n\times k}$, respectively, denote the cluster results of the link information and the attribute information, $\mathbf{U} \in \mathbb{R}_+^{k\times k}$ is a transition matrix used to describe the relationship between link information clusters and attribute information clusters, and $\|\mathbf{I}-\mathbf{U}\|_F^2$ and $\left\|\mathbf{U}\mathbf{1}_k^T-\mathbf{1}_k^T\right\|_F^2$ are the constraint items used to leverage the attribute information more effectively. The first two parts of RSECD play a leading role in the process of discovering communities, and the last three parts can adaptively guide the model to improve the performance as much as possible no matter whether the attribute information is good or poor. It is worth noting that the preassigned cluster numbers of the link information and the attribute information can be inconsistent. Therefore, RSECD is more flexible to exploit the attribute information with different qualities.

D. Multilayer Networks

In some real-world complex networks, the same nodes may have multiple types of relationships. For example, users in social networks can be connected via different types of relationships (e.g., classmate, profession, and family). Such networks are referred to as multilayer or multiplex networks, where every layer is a network represented a different semantic relationship among nodes. On the contrary, single-layer networks that are the most commonly used only have one type of relationship between nodes. Multilayer networks also have a community structure, but community detection in them aims to detect clusters of nodes shared by all layers, which makes it different from community detection in single-layer networks. Papalexakis *et al.* [48] concluded that if the network information in different layers can be fully exploited, we can detect better communities in multilayer networks than in single-layer networks.

Recently, the problem of community detection in multilayer networks has gained increasing interest and some methods have been continually proposed. A short related survey has been made in [49], but it does not cover many new representative methods, especially NMF-based methods. Due to the inherent advantages of information fusion, methods based on NMF demonstrate the superior performance. Unexceptionally, these methods all utilize the joint NMF framework to fuse these so-called multiview network data. For example, Gligorijević *et al.* [50] proposed a method named WSSNMTF, which is based on weighted simultaneous SNMF. This method first represents a multilayer network with N layers by a set of adjacency matrices $\mathbf{A}^{(i)} \in \mathbb{R}_+^{n\times n}$ $(i = 1, 2, \ldots, N)$ and meanwhile introduces a weight matrix $\mathbf{\Omega}^{(i)} \in \mathbb{R}_+^{n\times n}$ defined as follows to accelerate the later decomposition operations:

$$\Omega_{\text{ab}}^{(i)} = \begin{cases}1, & \text{if } v_a \text{ is connected with } v_b \text{ in the } i\text{th layer} \\ 0, & \text{otherwise.}\end{cases} \tag{14}$$

Then, the final community detection model is constructed by jointly factoring each $\mathbf{A}^{(i)}$ using NMTF with the common community indicator matrix $\mathbf{H}$

$$\begin{aligned}\min \mathcal{L}\left(\mathbf{H},\mathbf{S}^{(i)}\right) = {} & \sum_{i=1}^{N}\left\|\mathbf{\Omega}^{(i)} \circ \left(\mathbf{A}^{(i)}-\mathbf{H}\mathbf{S}^{(i)}\mathbf{H}^T\right)\right\|_F^2 \\ & + \sum_{i=1}^{N}\eta_i\left\|\mathbf{S}^{(i)}\right\|_1 \\ & \text{s.t.}\quad \mathbf{H}\geq 0, \mathbf{S}^{(i)}\geq 0\end{aligned} \tag{15}$$

where ℓ_1 norm of $\left\|\mathbf{S}^{(i)}\right\|$ imposes the sparsity constraint on $\mathbf{S}^{(i)}$ and η_i for $i \in \{1, 2, \ldots, N\}$ are tradeoff parameters of the corresponding constraint items.

As the extension of WSSNMTF, a general framework named NF-CCE [51] for extracting communities from multilayer networks using NMF is proposed. NF-CCE consists of two parts: 1) for each layer i, using SNMF to obtain its low-dimensional representation $\mathbf{H}^{(i)}$ under orthogonal constraints $\mathbf{H}^{(i)T}\mathbf{H}^{(i)} = \mathbf{I}$ $(\mathbf{H}^{(i)} \in \mathbb{R}_+^{n\times k})$ and 2) collectively decomposing all matrices $\mathbf{A}^{(i)}$ $(i = 1, 2, \ldots, N)$ into a common community indicator matrix $\mathbf{H}$ while enforcing $\mathbf{H}^{(i)}$ to be close enough to $\mathbf{H}$. The simplified objective function of NF-CCE is

$$\begin{aligned}\min \mathcal{L}(\mathbf{H},\mathbf{H}^{(i)}) = {} & \sum_{i=1}^{N}\left\|\mathbf{A}^{(i)}-\mathbf{H}\mathbf{H}^T\right\|_F^2 \\ & + \sum_{i=1}^{N}\left\|\mathbf{H}\mathbf{H}^T-\mathbf{H}^{(i)}\mathbf{H}^{(i)T}\right\|_F^2 \\ & \text{s.t.}\quad \mathbf{H}\geq 0, \mathbf{H}^{(i)}\geq 0, \mathbf{H}^{(i)T}\mathbf{H}^{(i)} = \mathbf{I}\end{aligned} \tag{16}$$

where the second part is the Grassmann manifold [52] constraint item applied to minimize the distance between $\mathbf{H}^{(i)}$ and $\mathbf{H}$. Experimental results show that this type of constraint improves the performance of NF-CCE greatly. Similar to WSSNMTF and NF-CCE, MTRD [53] and LJ-SNMF [54] are also using the joint NMF model with constraint items and both achieve good performance.

Compared with these aforementioned methods, another representative method S2-jNMF [55] has two unique features. First, it introduces multilayer modularity density to evaluate the performance of community detection, and meanwhile, it proves that the trace optimization of multilayer modularity density is equivalent to the objective functions of multiview clustering using NMF. This provides the solid theoretical foundation for designing joint NMF-based algorithms for community detection in multilayer networks. Second, it is a semisupervised method that considers prior information existing in multilayer networks. Specifically, S2-jNMF assumes that nodes in common dense subgraphs across all layers are more likely to be in the same community and hence constructs the corresponding quantization matrix $\mathbf{P} \in \mathbb{R}_+^{n\times n}$ to encode this prior information to guide the following community detection model:

$$\mathcal{L}\big(\mathbf{H}, \mathbf{F}^{(i)}\big) = \sum_{i=1}^{N} \big\|\mathbf{A}^{(i)} + \alpha\mathbf{P} - \mathbf{H}\mathbf{F}^{(i)T}\big\|_F^2 \quad \text{s.t.} \quad \mathbf{H} \geq 0, \mathbf{F}^{(i)} \geq 0 \tag{17}$$

where $\mathbf{F}^{(i)}$ $(i = 1, 2, \ldots, N)$ is treated as the coefficient matrix and α is the weight parameter. In general, S2-jNMF can serve as a general semisupervised framework for community detection in multilayer networks.

E. Dynamic Networks

Complex networks that have temporal features are called dynamic networks. Such networks can be divided into multiple network slices and they are pervasive in real world. For instance, online social networks can be modeled as dynamic networks because new users could constantly joint or quit as time goes by, while relationships between users are also constantly established or dismissed. Detecting communities from dynamic networks is very meaningful because it can track the evolutions, even detect mutations of communities. Furthermore, it can help us to understand and predict the development trends of networks.

Due to the dynamic property, community detection in dynamic networks is more challenging than in static networks. Rossetti and Cazabet [56] surveyed many methods specially developed to solve this problem, where NMF-based methods are popular. Generally, existing NMF-based methods for community detection in dynamic networks can be simply categorized into two types: online methods and offline methods. Online methods learn community structure at time t by explicitly utilizing information about the network topology and the community structure at time $t-1$. Formally, the basic form of online methods can be represented as the following objective function:

$$\begin{aligned}\min \mathcal{L}(\mathbf{H}_t, \mathbf{Q}_t) = {} & \big\|\mathbf{A}_t - \mathbf{H}_t\mathbf{H}_t{}^T\big\|_F^2 \\ & + \alpha\|\mathbf{H}_{t-1}\mathbf{Q}_t - \mathbf{H}_t\|_F^2 \\ & \text{s.t.} \quad \mathbf{H}_t \geq 0, \mathbf{Q}_t \geq 0\end{aligned} \tag{18}$$

where $\mathbf{A}_t$ and $\mathbf{H}_t$, respectively, denote the adjacency matrix and the community indicator matrix at time t and $\mathbf{Q}_t$ is a transition matrix used to depict the evolution relationships of communities at time $t-1$ and t. On the contrary, offline methods need to exploit information from all previous $t-1$ time slices and the basic form of them can be represented as follows:

$$\begin{aligned}\min \mathcal{L}(\mathbf{H}_t, \mathbf{Q}_t) = {} & \sum_{t=1}^{S} \big\|\mathbf{A}_t - \mathbf{H}_t\mathbf{H}_t{}^T\big\|_F^2 \\ & + \alpha\sum_{t=1}^{S}\|\mathbf{H}_{t-1}\mathbf{Q}_t - \mathbf{H}_t\|_F^2 \\ & \text{s.t.} \quad \mathbf{H}_t \geq 0, \mathbf{Q}_t \geq 0\end{aligned} \tag{19}$$

where S is the number of time slices. By comparing (18) with (19), it is easy to observe that: at time t, the community structure obtained by offline methods is affected by the data of all time slices, but for online methods, it is only effected by the data at time $t-1$. Essentially, online methods provide an incremental way for learning community structure over time, so they will perform more efficiently in most cases.

Following these two basic forms, some researchers devoted to developing more effective variants. For online methods, Ma and Dong [57] proposed a semisupervised evolutionary NMF (sE-NMF) method. Instead of $\mathbf{A}_t$, sE-NMF factorizes the following matrix $\widehat{\mathbf{A}}_t^*$ to obtain community structure:

$$\widehat{\mathbf{A}}_t^* = \mathbf{A}_t^* + \gamma\,\mathbf{Z}_t\mathbf{Z}_t{}^T \tag{20}$$

where $\mathbf{Z}_t$ contains priori information that nodes belong to the local communities at time t and $\mathbf{A}_t^*$ is a temporal smoothness matrix defined as

$$\mathbf{A}_t^* = \alpha\mathbf{A}_t + (1-\alpha)(\mathbf{A}_t - \mathbf{A}_{t-1}). \tag{21}$$

Based on sE-NMF, Ma *et al.* [58], [59] further developed two improved versions: GrENMF and Cr-ENMF, respectively, which both use graph regularized technique to achieve better performance. Similar methods also include ECGNMF [60] and DGR-SNMF [61]. Being different from these methods, Wang *et al.* [62] presented a method based on dynamic Bayesian nonnegative matrix factorization (DBNMF). This method obtains the community structure at time t by solving the following negative log posterior objective function:

$$\begin{aligned}\min \mathcal{L}\big(\mathbf{H}_t, \boldsymbol{\beta}_t\big) = {} & -\log P(\mathbf{A}_t|\mathbf{H}_t) - \log P\big(\mathbf{H}_t|\mathbf{H}'_{t-1}, \delta\big) \\ & -\log P\big(\mathbf{H}_t|\boldsymbol{\beta}_t\big) - \log P\big(\boldsymbol{\beta}_t\big)\end{aligned} \tag{22}$$

where δ is a fixed hyperparameter, $\mathbf{H}'_{t-1}$ is constructed by deleting the rows from $\mathbf{H}_{t-1}$ representing the nodes disappeared at time t and adding the rows of newly added nodes at time t, $P(\mathbf{A}_t|\mathbf{H}_t)$ and $P(\mathbf{H}_t|\mathbf{H}'_{t-1}, \delta)$ are computed using the Poisson distribution with Poisson rate $\sum_p^k h_{ip}h_{pj}^T$, and $P(\mathbf{H}_t|\boldsymbol{\beta}_t)$ and $P(\boldsymbol{\beta}_t)$ are, respectively, computed using the half-normal distribution with parameter $\boldsymbol{\beta}_t$ and the Gamma distribution. Similar to DBNMF, Márquez *et al.* [63] also proposed a method based on the Bayesian NMF. This method not only has the features of DBNMF but can also be applied to dynamic networks with node attributes.

In terms of offline methods, a few efforts have been made to obtain better performance. For example, methods proposed in [64] and [65] both impose ℓ_1 norm to $\mathbf{H}_i$ to obtain more clearer temporal community structure. Jiao *et al.* [66] presented a constrained common cluster-based method named C^3, whose objective function is denoted as

$$\min \mathcal{L}(\mathbf{W}, \mathbf{H}_t) = \lambda \sum_{i=1}^{S} \left\| \mathbf{P}_t - \mathbf{W}{\mathbf{H}_t}^T \right\|_F^2 + (1-\lambda) \sum_{t=1}^{S} \left\| \mathbf{A}_t - \mathbf{H}_t{\mathbf{H}_t}^T \right\|_F^2 \quad \text{s.t.} \quad \mathbf{W} \geq 0, \mathbf{H}_t \geq 0 \tag{23}$$

where $\mathbf{P}_t$ is the Markov steady-state matrix of the network at time t, $\mathbf{W}$ is the common cluster indicator matrix shared by all network slices, and λ is the weight parameter. In [67]–[69], researchers all proposed to conduct community detection task and link prediction task under the unified NMF-based framework, and experiment results show that this strategy can mutually reinforce the performance of every task at the same time.

These offline methods above only use the link information. Currently, there are some methods considering both the link and content information to improve the performance. Chimera [70] is one of the representative methods. It is also based on the consensus factorization model widely used in community detection in attributed networks. In particular, it introduces a temporal regularized term like $\sum_{t=1}^{S} \|\mathbf{H}_t - \mathbf{H}_{t-1}\|_2^F$ to ensure that the community structures between successive network slices do not change dramatically. This feature enables it to be effectively used in some dynamic networks, where nodes have stable links. For example, in coauthorship networks, authors in the same research team can maintain cooperative relationships for a long time, and likewise, research teams (i.e., communities) also remain relatively stable over time.

F. Large-Scale Networks

Some real-world complex networks (e.g., Facebook[3] and Tweet[4] online social networks) often have millions, even billions of nodes and links. Detecting communities from these large-scale networks needs highly efficient methods. Some other types of methods often have relatively idea time complexities, such as $O(|E|)$ in label propagation-based method LP-LPA [71] and $O(nlog(n))$ in hierarchical clustering-like-based method ECES [72]. However, most existing NMF-based methods are very time consuming. In Table II, we list the time complexities of some methods introduced above. It can be observed that they are all beyond $O(n^2)$, which makes them almost impossible to be applied to large-scale networks efficiently.

To solve the bottleneck problem in efficiency, some improved methods have been proposed. In general, these methods mainly include two types: methods with linear or near-linear time complexity, and parallel and distributed methods. Among the first type of methods, BIGCLAM presented in [73] is very representative because it is the first to solve the efficiency problem of NMF-based methods for community detection. Unlike many methods using squared Frobenius norm, BIGCLAM employs log likelihood to construct its objective function

$$\min \mathcal{L}(\mathbf{H}) = -\sum_{(u,v)\in E} \log\left(1 - \exp\left(-\mathbf{H}_{u\cdot}\mathbf{H}_{v\cdot}^T\right)\right) + \sum_{(u,v)\notin E} \mathbf{H}_{u\cdot}\mathbf{H}_{v\cdot}^T \quad \text{s.t.} \quad \mathbf{H} \geq 0. \tag{24}$$

To efficiently solve the optimization problem in (24), BIGCLAM adopts an improved block coordinate gradient descent algorithm presented in [74], which helps it run 10–100 times faster than competing approaches on benchmark large-scale networks. Du *et al.* [75] presented a hierarchical community detection method HierSymNMF2 based on rank-2 NMF. HierSymNMF2 introduces a divide-and-conquer strategy to boost the efficiency and is composed of an iterative process with two stages: choosing one of the communities to split and splitting the chosen community into two communities. The task of splitting a community is performed by the rank-2 NMF model shown as follows:

$$\min \mathcal{L}(\mathbf{W}, \mathbf{H}) = \left\| \mathbf{S} - \mathbf{W}\mathbf{H}^T \right\|_F^2 + \alpha \|\mathbf{W} - \mathbf{H}\|_F^2, \quad \text{s.t.} \quad \mathbf{W} \geq 0, \mathbf{H} \geq 0 \tag{25}$$

where $\mathbf{S} \in \mathbb{R}_+^{n\times n}$ is a similarity matrix representing G, and the ranks of $\mathbf{W}$ and $\mathbf{H}$ are both 2, i.e., $k = 2$. This objective function can be solved by alternatively optimizing the following subproblems for $\mathbf{W}$ and $\mathbf{H}$:

$$\min_{\mathbf{W}\geq 0} \left\| \begin{bmatrix} \mathbf{H} \\ \sqrt{\alpha}\mathbf{I}_2 \end{bmatrix} \mathbf{W}^T - \begin{bmatrix} \mathbf{S} \\ \sqrt{\alpha}\mathbf{H}^T \end{bmatrix} \right\|_F^2 \tag{26}$$

$$\min_{\mathbf{H}\geq 0} \left\| \begin{bmatrix} \mathbf{W} \\ \sqrt{\alpha}\mathbf{I}_2 \end{bmatrix} \mathbf{H}^T - \begin{bmatrix} \mathbf{S} \\ \sqrt{\alpha}\mathbf{W}^T \end{bmatrix} \right\|_F^2 \tag{27}$$

where $\mathbf{I}_2$ is the 2×2 identity matrix. In particular, (26) and (27) can be efficiently solved by an improved active-set-type algorithm described in [76]. Experimental results show that HierSymNMF2 has a good scalability and even performs better than BIGCLAM. By analyzing BIGCLAM and HierSymNMF2, we find that they both utilize more efficient optimization algorithms for NMF-based community detection model to improve the final efficiency. Therefore, to improve the efficiency, some efficient algorithms specially designed for NMF can also be adopted in NMF-based methods for community detection, such as cyclicCDSymNMF [77], ℓ1-GNMF [78], and OGNMF [79].

To obtain higher efficiency, some parallel and distributed methods are proposed. These methods can make full use of computing resources of machines to greatly speed up the processing on large-scale networks. For example, in [80] and [81], we, respectively, proposed methods DRNMFSR and TCB. They both implement the iterative update rules using the MapReduce distributed computing framework. Specifically, for maximizing parallelism, they first divide every iterative update rule into multiple stages and then further divide every stage into multiple components. Finally, every component is implemented using multiple continuous MapReduce jobs. Fig. 3 shows the entire flowchart of updating $\mathbf{H}$ in the TCB method. We can refer to [81] for more details.

[3]http://www.facebook.com

[4]http://www.twitter.com

TABLE II
TIME COMPLEXITIES OF NMF-BASED METHODS FOR COMMUNITY DETECTION (NOTE THAT HERE, WE OMIT THE NUMBER OF ITERATIONS)

Category	Method	Time complexity	Remark
Topology networks	NMTF [21]	$O(n^2k)$	
	BNMTF [22]	$O(n^2k + nk^2)$	
	PCSNMF [23]	$O(n^2k + l^2)$	l is the number of labeled nodes
	PSSNMF [24]	$O(n^2k + nk^2)$	
	HPNMF [25]	$O(n^2k + nk^2)$	
Signed networks	JNMF [31]	$O(n^2k + nk^2)$	
	SGNMF [32]	$O(n^2k)$	
	MCNMF [33]	$O(n^2k)$	
	ReS-NMF [36]	$O(n^2k + nk^2 + k^3)$	
	BRSNMF [37]	$O(n^2k + nk^2 + k^3)$	
Attributed networks	FSL [40]	$O((mn + n^2)k)$	m is the length of the attribute set
	JWNMF [41]	$O((n^2 + m^2 + mn)k)$	
	NMTFR [42]	$O((nw + nm + mw + n^3 + w^2)k)$	w is the number of messages
	CFOND [43]	$O(mn(k + c) + m^2c + n^2k)$	c is the number of feature clusters
	SCI [44]	$O(mnk + n^2k)$	
	DII [46]	$O(nmk + n^2k + n^2m)$	
	RSECD [47]	$O(n^2k + 2mnk + nk^2)$	
Multi-layer networks	WSSNMTF[50]	$O(N(n^2k + nk^2))$	N is the number of layers
	NF-CCE [51]	$O(N(n^2k + nk^2))$	
	MTRD [53]	$O(Nn^2k)$	
	LJ-SNMF [54]	$O(N(n^2k + nk^2))$	
	S2-jNMF [55]	$O(Nn^2k)$	
Dynamic networks	sE-NMF [57]	$O(S(n^3 + n^2k))$	S is the number of networks slices
	GrENMF [58]	$O(Sn^2k)$	
	Cr-ENMF [59]	$O(Sn^2k)$	
	ECGNMF [60]	$O(Sn^2k)$	
	DGR-SNMF [61]	$O(Sn^2k)$	
	DBNMF [62]	$O(Sn^2k)$	
	C^3 [66]	$O(S(n^2k + nk^2) + mk)$	
	Chimera [70]	$O(S(n^2k + nmk))$	

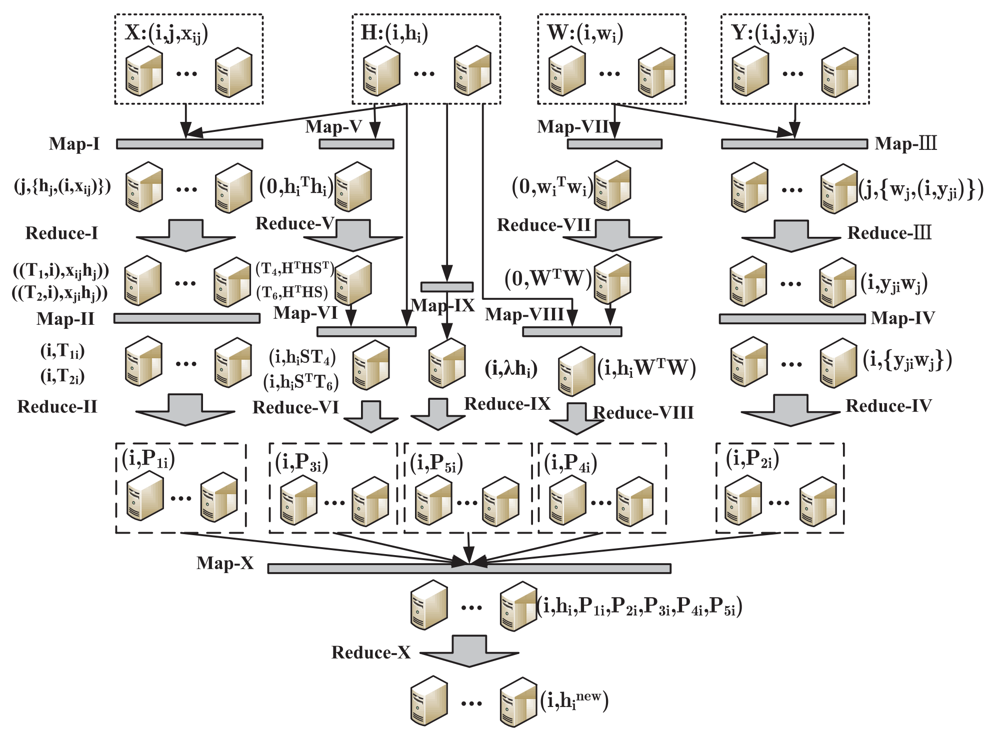

Fig. 3. Updating **H** on MapReduce clusters [81].

Similar to DRNMFSR and TCB, NMF variants presented in [82]–[84] can be implemented using some distributed computing frameworks, including iMapReduce, MPI, and GPU, so they can also be used to detect communities from large-scale networks efficiently. Comparing with methods with linear or near-linear time complexity, these parallel and distributed methods have more advantages in terms of dealing with large-scale networks. After all, they can use more or even unlimited computing and storage resources of the machine clusters.

V. Common Problems and Solutions

In this section, we summarize several problems that are common to NMF-based methods for community detection. Actually, many other types of methods may also encounter similar problems. We also introduce representative solutions to these problems. Note that some solutions are completely from related works on NMF and do not specially focus on the topic of community detection.

A. Initialization Method

Before iteratively solving the NMF-based community detection model, many methods initial factor matrices (e.g., $\mathbf{W}$ and $\mathbf{H}$) by generating all their entries uniformly at random. Although this way is simple and straightforward, it often leads to much slower convergence and unstable results. To solve this problem, some advanced initialization methods have been developed. They are mainly divided into the following three types.

1) *SVD Based:* Boutsidis and Gallopoulos [85] first proposed an SVD-based initialization method named NNDSVD for NMF. We can employ this method to initialize every NMF-based community detection model. For example, for the SNMF model for community detection like $\mathbf{X} \approx \mathbf{H}\mathbf{H}^T$, $\mathbf{H}$ can be initialized by using the following steps. First, factorize $\mathbf{X}$ into the form of SVD: $\mathbf{X} = \Phi\Sigma\Phi^T$, where $\Sigma = \mathrm{diag}(\delta_1, \delta_2, \ldots, \delta_k)$ contains all feature values of $\mathbf{X}$ and $\delta_1 \geq \delta_2 \geq \ldots \geq \delta_k > 0$. Second, initialize the first column of $\mathbf{H}$: $\mathbf{H}_{.1} = (\delta_1)^{1/2}\Phi_{.1}$. Third, for $\forall \mathbf{H}_{.j} \in \mathbf{H}$ $(j = 2, \ldots, k)$, first compute its positive matrix $[\mathbf{H}_{.j}]^+$ and negative matrix $[\mathbf{H}_{.j}]^-$ (related definitions are mentioned in [85]) and then initialize it by the following rule:

$$\mathbf{H}_{.j} = \begin{cases} \sqrt{\delta_j}[\mathbf{H}_{.j}]^+, & \text{if } \left\|[\mathbf{H}_{.j}]^+\right\|_1 \geq \left\|[\mathbf{H}_{.j}]^-\right\|_1 \\ \sqrt{\delta_j}[\mathbf{H}_{.j}]^-, & \text{otherwise.} \end{cases} \tag{28}$$

NNDSVD has been proven to be effective in enhancing the convergence rate and stability of NMF. Similar methods include SVDCNMF [86] and NNSVD-LRC [87]. They are both improved versions of NNDSVD and can be expected to perform better when used in initializing the NMF-based community detection model.

2) *Clustering Based:* Because NMF itself is a good clustering model, it is natural to use clustering methods to initial NMF. For example, Wild *et al.* [88] proposed to use k-means and spherical k-means clustering results (i.e., clusters centroids and cluster indicator matrix) to initialize NMF. Specifically, every column of $\mathbf{W}$ is initialized by the corresponding cluster centroid vector and $\mathbf{H}$ is directly initialized by the cluster indicator matrix. Besides, fuzzy c-means and subtractive clustering methods have also been proved to be effective in initializing NMF [89]–[91].

3) *Swarm Intelligence Based.* Recently, swarm intelligence algorithms have been used to initialize NMF. They generally operate on a population of optimization solutions in the search space. Janecek and Tan [92] investigated the effectiveness of five swarm intelligence algorithms in terms of initializing NMF, including genetic algorithms, particle swarm optimization, fish school search, differential evolution, and fireworks algorithm. Experimental results show that they are well suited as initialization enhancers of NMF. It should be noted that swarm intelligence algorithms all have high computational costs, but they can be implemented in parallel.

B. Constructing More Informative Feature Matrix

We can observe that most of the aforementioned NMF-based methods for community detection select the adjacency matrix $\mathbf{A}$ as the feature matrix $\mathbf{X}$ used for factorization. Although $\mathbf{A}$ is the most available, it is not informative enough due to the reason that it is often very sparse and only represents the local structure features. Undoubtedly, if the feature matrix cannot contain enough information, it will degrade the performance of community detection.

Constructing more informative feature matrix to replace $\mathbf{A}$ has become one of the focuses in many existing NMF-based methods for community detection. Recently, many effective solutions have been proposed, among which some use high-order proximity matrix as the feature matrix. For example, Wang *et al.* [93] first treated $\mathbf{A}$ as the first-order proximity matrix $\mathbf{S}'$ and then defined the second-order proximity matrix $\mathbf{S}'' = [\mathbf{S}''_{ij}]^{n\times n}$ based on the metric of cosine similarity

$$\mathbf{S}''_{ij} = \frac{\mathbf{S}'_{i.} \cdot \mathbf{S}'_{j.}}{\left\|\mathbf{S}'_{i.}\right\| \left\|\mathbf{S}'_{j.}\right\|}. \tag{29}$$

Finally, to preserve both the first- and second-order proximities, they use $\widehat{\mathbf{S}} = \mathbf{S}' + \eta\mathbf{S}''$ as the final proximity matrix, where η is the weight parameter. Zhang and Zhou [94] also adopted the same proximity matrix as the feature matrix. Besides the node proximity matrix, some solutions use a node similarity matrix computed by the special node similarity measure, such as SimRank node similarity matrix used in [80] and Random walk transition probability matrices used in [95] and [96]. Experimental results show that not only high-order proximity matrix but also node similarity matrix can capture more local and global structure features. Moreover, using them as the feature matrices all can boost the performance of NMF-based community detection to some extent.

Being different from the solutions above only using the link information, some other solutions propose to produce a new feature matrix by integrating extra information. For example, Ma *et al.* [97] constructed the new feature matrix $\mathbf{X}'$ by adding must-link constraints matrix $\mathbf{M}_{ml}$ and cannot-link constraints matrix $\mathbf{M}_{cl}$ to $\mathbf{X}$, which could be $\mathbf{A}$ or a node similarity matrix

$$\mathbf{X}' = \mathbf{X} - \alpha\mathbf{M}_{ml} + \beta\mathbf{M}_{cl} \tag{30}$$

where α and β are real numbers small enough to ensure that $\mathbf{X}'$ is positively definite. Li *et al.* [98] also follow the similar idea, but they choose to merge node content information into the random walk transition probability matrix to construct $\mathbf{X}'$.

It is worth noting that some solutions have tried to use a low-dimensional network embedding matrix as the new feature matrix. For example, Lin *et al.* [99] utilized node2vec [100] to produce network embedding matrix and factorize this matrix using NMF to obtain community structures. Experimental results show that this way not only reduces the time complexity of NMF-based community detection but also helps to improve the performance of community detection greatly. Due to the popularity and powerful ability to learn node representation, some classical graph embedding methods, such as Deepwalk [101] and LANE [102], will also be expected to be good choices used to construct a more informative feature matrix.

C. Determining k Automatically

Most existing NMF-based community detection methods need to preassign the number of communities k. However, it is often very difficult to set an accurate k value manually, especially when dealing with large-scale complex networks without ground-truth communities. To address this problem, currently, some methods have been proposed to determine k automatically, including methods based on minimizing residual error, methods based on automatic relevance determination (ARD), methods based on matrix spectrum theory, and methods based on instability.

Methods based on minimizing residual error provide a straightforward way to obtain the optimal k. For example, the method proposed in [103] first treats k as an independent variable and evaluates the residual error of the objective function with respect to each possible value of k and finally selects the optimal k that leads to the lowest residual error. Methods based on ARD are most widely used [62], [104]–[106]. In this type of method, k is set to be a large value at first, and then, irrelevant vectors in $\mathbf{W}$ and $\mathbf{H}$ are pruned via certain probabilistic inference framework (e.g., Bayesian model). Finally, k is set to be the number of remaining relevant vectors in $\mathbf{W}$ and $\mathbf{H}$.

Differing from methods above, methods based on matrix spectrum theory often use the error between $\mathbf{A}$ and the product of its eigenvalues and eigenvectors to select appropriate k. For example, Ma and Dong [57] defined the spectrum of $\mathbf{A}$ as the set of the eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$ and the corresponding eigenvectors $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n$ and selected the k satisfying the rule

$$k = \arg\min_{k} \sqrt{\frac{\left\| \sum_{i=1}^{k} \lambda_i \mathbf{a}_i \mathbf{a}_i^T \right\|}{\|\mathbf{A}\|}} > \delta \tag{31}$$

where δ is used to control the approximation. A similar method can be found in [107]. Method based on instability was first proposed in [108]. Its basic operation process is: for the candidate k, we first run NMF with random initial values to factorize $\mathbf{A}$ τ times. This will produce τ basis matrices: $\mathbf{W}_1, \mathbf{W}_2, \ldots, \mathbf{W}_\tau$. For any two matrices $\mathbf{W}_i$ and $\mathbf{W}_j$, we define a matrix $\mathbf{R} = [\mathbf{R}_{\mathrm{ab}}]^{k\times k}$, where $\mathbf{R}_{\mathrm{ab}}$ denotes the cross correlation between the ath column of $\mathbf{W}_i$ and the bth column of $\mathbf{W}_j$. Next, we define the dissimilarity between $\mathbf{W}_i$ and $\mathbf{W}_j$ as

$$\mathrm{diss}\big(\mathbf{W}_i, \mathbf{W}_j\big) = \frac{1}{2k}\left(2k - \sum_{a=1}^{k} \max_{1\le b\le k} \mathbf{R}_{\mathrm{ab}} - \sum_{b=1}^{k} \max_{1\le a\le k} \mathbf{R}_{\mathrm{ab}}\right). \tag{32}$$

The instability for k is computed by the average dissimilarity of all $\tau(\tau-1)/2$ pairs of basis matrices

$$\gamma(k) = \frac{2}{\tau(\tau-1)} \sum_{1\le i<j\le\tau} \mathrm{diss}(\mathbf{W}_i, \mathbf{W}_j). \tag{33}$$

Finally, by repeating the process above for each candidate k, k corresponding to minimal $\gamma(k)$ is selected as the final number of communities. In [55] and [58], experiment results show that methods based on instability are very effective in determining k automatically.

D. Overlapping Community Detection

Overlapping communities are common in complex networks. They allow any node to be assigned to multiple communities. Because of the inherent soft clustering ability, NMF is very suitable to be used to detect overlapping communities. Algorithm 1 and Fig. 2 both depict the general process of NMF-based overlapping community detection, in which the setting of threshold ϕ is crucial. Recently, most methods specify ϕ manually, but it is not easy to find a proper ϕ. Moreover, an improper ϕ may result in totally wrong community membership assignments.

To avoid setting ϕ manually, seeking for a binary community indicator matrix $\mathbf{H}$ is an ideal solution. In such $\mathbf{H}$, all elements are 0 or 1. If $\mathbf{H}_{ij}$ is 1, then v_i can be explicitly assigned into C_j without any strength threshold judgments. At present, there are two methods proposed to achieve this goal: SBMF [109] and discrete NMF [110]. In SBMF, the Heaviside step function is skillfully embedded into the SNMF model to learn the binary $\mathbf{H}$

$$\min \mathcal{L}(\mathbf{H}, h^*) = \left\|\mathbf{A} - f(\mathbf{H}-h^*) f(\mathbf{H}-h^*)^T\right\|_F^2, \quad \text{s.t.} \quad \mathbf{H} \ge 0. \tag{34}$$

In (34), h^* is also a threshold, but it can be learned automatically by optimizing $\mathcal{L}(\mathbf{H}, h^*)$. $\mathbf{H} - h^*$ is elementwise operation and $f(x)$ is the Heaviside step function defined as

$$f(x) = \begin{cases} 1, & \text{if } x > 0 \\ 0, & x \le 0. \end{cases} \tag{35}$$

Once h^* is determined, the final binary $\mathbf{H}$ can be obtained via $\mathbf{H} = f(\mathbf{H} - h^*)$. In discrete NMF, its binary strategy is different from SBMF. Specially, it introduces a rotation matrix Ψ to smoothly transform the continuous $\mathbf{H}$ to the binary $\mathbf{H}'$ and its corresponding objective function is designed as

$$\min \mathcal{L}(\mathbf{H}, \mathbf{H}', \Psi) = \left\|\mathbf{A} - \mathbf{H}\mathbf{H}^T\right\|_F^2 + \alpha\left\|\mathbf{H} - \mathbf{H}'\Psi\right\|_F^2 \quad \text{s.t.} \quad \mathbf{H} \ge 0 \wedge \Psi\Psi^T = \mathbf{I}_k \wedge \mathbf{H}' \in \mathcal{H} \tag{36}$$

where $\mathcal{H} = \{\mathbf{H}' | \mathbf{H}' \in \{0,1\}^{n\times k} \wedge \mathbf{H}'\mathbf{1}_k \ge \mathbf{1}_n\}$ denotes the solution space of $\mathbf{H}'$ and α is a tradeoff parameter. By setting the orthogonality constraint on Ψ and minimizing $\mathcal{L}(\mathbf{H}, \mathbf{H}', \Psi)$, $\mathbf{H}\mathbf{H}'^T$ can approximate to $\mathbf{A}$, which makes binary $\mathbf{H}'$ also have good ability to uncover community structure.

VI. Future Research Directions

A. Constructing Non-Frobenius Norm-Based Objective Function

As mentioned above, most existing NMF-based methods for community detection select the squared Frobenius norm to construct the objective function. Although the squared Frobenius norm is a simple yet effective way to measure the cost of NMF, it has been proven that this norm is not robust against noises and outliers [111], which often occurs in real-world complex networks. In [112], we applied the $\ell_{2,1}$ norm to construct the objective function of SNMF for community detection: $\|\mathbf{A} - \mathbf{H}\mathbf{H}^T\|_{2,1}$. Experimental results show that this type of objective function can improve the performance, especially when dealing with complex networks containing nonnegligible noises.

Actually, the flexibility in selecting the objective cost functions is one of the advantages of NMF. Zhang [14] introduced several kinds of divergence functions that are applicable to measuring the cost of NMF, including Csisźar's-divergence, α-divergence, β-divergence, Bregman divergence, Itakura–Saito divergence, and KL divergence. Meanwhile, the corresponding solutions to the objective functions using different divergences are also given. Some of these divergences have demonstrated their superiorities over the Frobenius norm in some tasks, such as audio source separation [113] and image clustering [114]. These works give us a good inspiration that using these divergences to construct the objective function can also be expected to improve the performance of NMF-based community detection model.

By investigating existing works for NMF-based community detection, we find that a few of them are focusing on constructing the objective function, which is not based on the Frobenius norm. Therefore, aiming at the problem of community detection in different types of complex networks, in the future, there is still room for further exploring how to construct a more suitable objective function.

B. Community Detection in Heterogeneous Networks

In many related works in the literature, complex networks are often modeled as homogeneous networks, which have the same type of nodes and links. However, now, it is generally believed that it is more reasonable to model real-world complex networks as heterogeneous networks with different types of nodes and links. Compared to homogeneous networks, heterogeneous networks provide more information and contain richer semantics in nodes and links. This promotes the development of many heterogeneous networks mining tasks, including community detection focused in this article.

Detecting communities in heterogeneous networks is more difficult than in homogeneous networks because it needs to consider how to integrate various information to obtain high-quality communities. Shi *et al.* [115] reviewed related works on community detection in heterogeneous networks. We find that there are few works using NMF to detect communities from heterogeneous networks. However, we think that the information fusion ability reflected in community detection in attributed networks and multilayer networks makes NMF also suitable for detecting communities in heterogeneous networks. Actually, if we treat attributes in attributed networks as the nodes, then attributed networks can be modeled as heterogeneous networks, and hence, many NMF-based methods for community detection in attributed networks can be regarded as the methods for community detection in heterogeneous networks to some extent.

Generally speaking, it is very valuable to deeply study how to utilize NMF to detect communities from heterogeneous networks, which can further extend the application scope of NMF-based methods for community detection. Due to the heterogeneity, using NMF to detect communities from heterogeneous networks will face some challenging problems. One problem is that the imbalances of different kinds of information raise higher requirements for the information fusion ability of NMF-based community detection model. In view of this, introducing the adaptive information fusion mechanism will be a good direction. Besides, designing the special feature fusion engine such as EigFuse [116] is also meaningful. Another problem is that multiple types of nodes coexisting in a network lead to a new community paradigm: a community may include different types of nodes sharing the same topic. However, almost all existing NMF-based methods for community detection can only be used for uncovering communities, including the same type of nodes. To solve this problem, it will be a good solution to make full use of the coclustering ability of joint NMF, which has been demonstrated in [117].

C. Combining Deep Learning to Further Boost the Performance

As stated in Section III-A, NMF is a linear model. Some works have pointed out that the linear community detection model may be less effective when facing complex networks with various nonlinear features [118], [119]. Therefore, it is still necessary to further boost the performance of NMF-based community detection.

It is well known that deep learning has been widely used in many fields and shows superior performance. The major advantage of deep learning is that it can learn the task-friendly data representation. This enables it often to be used to boost the performance of shallow vector-based machine learning models, especially some clustering models such as k-means [120], spectral clustering [121], and fuzzy clustering [122]. Existing works show that the combination of these classical clustering algorithms and deep learning can generally boost the clustering performance.

As discussed in Section III-B, NMF is a clustering model in essence, so it can also be expected to combine deep learning to further boost the performance. At present, there have some related works, where the most representative one is deep NMF (DNMF) proposed by Trigeorgis *et al.* [123]. The principal idea of DNMF is to stack single-layer NMF into N $(N > 1)$ layers, thereby to obtain hierarchical mappings $(\mathbf{W}_1, \mathbf{W}_2, \ldots, \mathbf{W}_N)$ and corresponding data representations $(\mathbf{H}_1, \mathbf{H}_2, \ldots, \mathbf{H}_N)$. This hierarchical factorization model and the intuitive comparison between NMF and DNMF are,

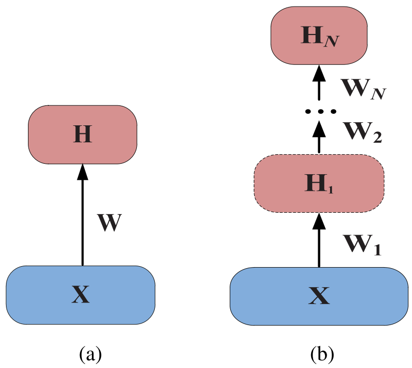

Fig. 4. Difference between (a) NMF and (b) DNMF.

respectively, shown in (37) and Fig. 4

$$\begin{aligned}\mathbf{X} &\approx \mathbf{W}_1\mathbf{H}_1\\ \mathbf{H}_1 &\approx \mathbf{W}_2\mathbf{H}_2\\ &\vdots\\ \mathbf{H}_{N-1} &\approx \mathbf{W}_N\mathbf{H}_N.\end{aligned} \tag{37}$$

With the aid of the multilayer factorization structure, DNMF becomes a deep model. This helps it obtain better data representation and clustering performance than shallow NMF model [124], [125]. Considering that complex networks contain hierarchical and structural information, such as node-level similarity and community-level similarity, currently some researchers have begun to apply DNMF to community detection. For example, methods proposed in [94], [126], and [127] are all based on DNMF and perform better than shallow NMF-based ones.

In general, these existing works on DNMF are just preliminary. We argue that they do not fully utilize the power of deep learning to boost the performance, and there are still many problems to be further studied, mainly including the following.

1) The mapping of DNMF in every layer belongs to the linear transformation, so DNMF is still a linear model as NMF. In view of this, as deep learning methods are based on neural networks, introducing nonlinear mapping functions to make DNMF a true nonlinear model can be expected to further improve its performance. Of course, this will undoubtedly increase the complexity of model optimization.
2) Recently, community detection methods based on DNMF are successfully used in topology networks, but the effectiveness of DNMF has not been verified in more complicated networks, including attributed networks, multilayer networks, and dynamic networks. Therefore, more DNMF variants applicable to these types of networks still need to be further explored.
3) DNMF does not use neural networks to achieve the goal of deep learning, so combining various neural networks with NMF is quite attractive. Zhang *et al.* [128] made the first attempt. They devised a nonlinear NMF model N-GNMF using neural networks. Formally, this model is denoted as the following objective function:

$$\min \mathcal{L}(\mathbf{W},\mathbf{H}) = \left\| f(\mathbf{X}) - \mathbf{W}\mathbf{H}^T \right\|_F^2 \quad \text{s.t.} \quad \mathbf{W} \geq 0, \mathbf{H} \geq 0 \tag{38}$$

where $f(\mathbf{X})$ stands for the low-dimensional representation of $\mathbf{X}$ obtained by given neural networks, such as deep autoencoder and deep convolution networks. N-GNMF performs better than NMF in the image clustering task. If applying it to community detection, we think that it can be further developed from two aspects: one is constructing a unified objective function with respect to NMF and neural networks. This can be expected to obtain more accurate community membership representation $\mathbf{H}$ by joint training. The other is replacing general neural networks with graph neural networks, which are specially designed to deal with graph data. We can deeply explore the combination strategy of NMF and some emerging graph neural networks, such as graph convolution network (GCN) [129], graph attention network (GAT) [130], graph autoencoder (GAE) [131], and others summarized in [132].

TABLE III

FEATURE STATISTICS OF EXISTING NMF-BASED COMMUNITY DETECTION METHODS (HERE, WE CALL THE FEATURE THAT CAN DETECT DYNAMIC COMMUNITIES AS DYNAMICITY)

Method	Fusion capability	Scalability	Dynamicity
NMTF [21]	×	×	×
BNMTF [22]	×	×	×
PCSNMF [23]	√	×	×
PSSNMF [24]	√	×	×
HPNMF [25]	√	×	×
HNMF [26]	√	×	×
A^2NMF [27]	√	×	×
PNMF [28]	√	×	×
JNMF [31]	×	×	×
SGNMF [32]	×	×	×
MCNMF [33]	×	×	×
ReS-NMF [36]	×	×	×
BRSNMF [37]	×	×	×
SPOCD [38]	×	×	×
FSL [40]	√	×	×
JWNMF [41]	√	×	×
NMTFR [42]	√	×	×
CFOND [43]	√	×	×
SCI [44]	√	×	×
ASCD [45]	√	×	×
DII [46]	√	×	×
RSECD [47]	√	×	×
WSSNMTF [50]	√	×	×
NF-CCE [51]	√	×	×
MTRD [53]	√	×	×
LJ-SNMF [54]	√	×	×
S2-jNMF [55]	√	×	×
sE-NMF [57]	×	×	√
GrENMF [58]	×	×	√
Cr-ENMF [59]	×	×	√
ECGNMF [60]	×	×	√
DGR-SNMF [61]	×	×	√
DBNMF [62]	×	×	√
C^3 [66]	√	×	√
Chimera [70]	√	×	√
BIGCLAM [73]	×	√	×
HierSymNMF2 [75]	×	√	×
cyclicCDSymNMF [77]	×	√	×
OGNMF [79]	×	√	×
DRNMFSR [80]	√	√	×
TCB [81]	√	√	×

D. Versatile Community Detection Methods

Many real-world networks, especially online social networks are large-scale, also contain multiple types of information (e.g., user relationships, user profiles, and posts in Facebook and Twitter online social networks). Besides, they are always evolving: nodes and links constantly appear or disappear, and other types of information are also always changing. These networks are often modeled as large-scale dynamic heterogeneous networks, which have the 4V characteristics of big data: value, volume, variety, and velocity.

Community detection in large-scale dynamic heterogeneous networks is more challenging. This needs to design a method that not only has good information fusion capability and scalability but can also detect dynamic communities. In Table III, we summarize the statistics of existing NMF-based community detection methods with these features. It can be clearly observed that none of the methods has all these features at the same time. Therefore, it will be promising to develop such versatile NMF-based community detection methods. To this end, integrating features of existing methods for heterogeneous networks, dynamic networks, and large-scale networks to design new methods applicable to large-scale dynamic heterogeneous networks is a good direction. Besides, to improve the scalability, we can explore to use incremental NMF (INMF) [133] to replace or cooperate with distributed NMF. INMF has been proven very efficient in some clustering tasks [134], [135], but it has not been applied to community detection yet.

VII. Conclusion

NMF has become a widely used model for community detection in complex networks and lots of related works have been continually presented. In this article, we conduct a comprehensive overview of existing NMF-based methods for community detection. From the perspective of network types that methods are applicable to, we group the existing methods into six categories: topology networks, signed networks, attributed networks, multilayer networks, dynamic networks, and large-scale networks. We first deeply review representative methods in every category and then introduce the common problems and their potential solutions. Finally, we point out four promising research topics.

Through this survey, we can fully understand the advantages of NMF for community detection summarized in Section I. Meanwhile, we can also realize that NMF for community detection still has some disadvantages. The most obvious ones are high time complexity and linear model. Fortunately, there have some successful attempts to overcome these drawbacks (e.g., some methods introduced in Section IV-F and Section VI-C). We believe that this survey can well serve as a valuable reference for researchers who are interested in NMF-based community detection and inspire them to devote more efforts to making NMF-based community detection more versatile.

References

[1] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proc. Nat. Acad. Sci. USA*, vol. 99, no. 12, pp. 7821–7826, Apr. 2002.

[2] S. Fortunato and D. Hric, "Community detection in networks: A user guide," *Phys. Rep.*, vol. 659, pp. 1–44, Nov. 2016.

[3] Y. van Gennip *et al.*, "Community detection using spectral clustering on sparse geosocial data," *SIAM J. Appl. Math.*, vol. 73, no. 1, pp. 67–83, Jan. 2013.

[4] Y. Li, K. He, K. Kloster, D. Bindel, and J. Hopcroft, "Local spectral clustering for overlapping community detection," *ACM Trans. Knowl. Discovery Data*, vol. 12, no. 2, pp. 1–27, Mar. 2018.

[5] E. Abbe, "Community detection and stochastic block models: Recent developments," *J. Mach. Learn. Res.*, vol. 18, pp. 1–86, Jan. 2018.

[6] S. E. Garza and S. E. Schaeffer, "Community detection with the label propagation algorithm: A survey," *Phys. A, Stat. Mech. Appl.*, vol. 534, Nov. 2019, Art. no. 122058.

[7] A. Jonnalagadda and L. Kuppusamy, "A survey on game theoretic models for community detection in social networks," *Social Netw. Anal. Mining*, vol. 6, pp. 1–24, Sep. 2016.

[8] H. J. Li, Z. Bu, Z. Wang, and J. Cao, "Dynamical clustering in electronic commerce systems via optimization and leadership expansion," *IEEE Trans. Ind. Informat.*, vol. 16, no. 8, pp. 5327–5334, Aug. 2020.

[9] F. Liu *et al.*, "Deep learning for community detection: Progress, challenges and opportunities," in *Proc. 29th Int. Joint Conf. Artif. Intell. (IJCAI)*, Jul. 2020, pp. 4981–4987.

[10] 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, Oct. 1999.

[11] F. Zhu and P. Honeine, "Biobjective nonnegative matrix factorization: Linear versus kernel-based models," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 7, pp. 4012–4022, Jul. 2016.

[12] Z. Yang and E. Oja, "Linear and nonlinear projective nonnegative matrix factorization," *IEEE Trans. Neural Netw.*, vol. 21, no. 5, pp. 734–749, May 2010.

[13] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Proc. 15th Int. Conf. Neural Inf. Process. Syst. (NIPS)*, Dec. 2001, pp. 535–541.

[14] Z. Y. Zhang, "Nonnegative matrix factorization: Models, algorithms and applications," in *Data Mining, Foundations and Intelligent Paradigms*, vol. 24. Berlin Germany: Springer, 2012, pp. 99–134.

[15] W. S. Ng and W. W. Tan, "Some properties of various types of matrix factorization," in *Proc. ITM Web Conf.*, vol. 36, Jan. 2021, Art. no. 03003.

[16] Z. Shu *et al.*, "Rank-constrained nonnegative matrix factorization for data representation," *Inf. Sci.*, vol. 528, pp. 133–146, Aug. 2020.

[17] E. Esser, M. Moller, S. Osher, G. Sapiro, and J. Xin, "A convex model for nonnegative matrix factorization and dimensionality reduction on physical space," *IEEE Trans. Image Process.*, vol. 21, no. 7, pp. 3239–3252, Jul. 2012.

[18] X. Luo, M. Zhou, Y. Xia, and Q. Zhu, "An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems," *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 1273–1284, May 2014.

[19] C. Ding, X. He, and H. D. Simon, "On the equivalence of nonnegative matrix factorization and spectral clustering," in *Proc. SIAM Int. Conf. Data Mining (SDM)*, Apr. 2005, pp. 606–610.

[20] I. Psorakis, S. Roberts, M. Ebden, and B. Sheldon, "Overlapping community detection using Bayesian non-negative matrix factorization," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 83, Jun. 2011, Art. no. 066114.

[21] F. Wang, T. Li, X. Wang, S. Zhu, and C. Ding, "Community discovery using nonnegative matrix factorization," *Data Mining Knowl. Discovery*, vol. 22, no. 3, pp. 493–521, 2011.

[22] Y. Zhang and D.-Y. Yeung, "Overlapping community detection via bounded nonnegative matrix tri-factorization," in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2012, pp. 606–614.

[23] X. Shi, H. Lu, Y. He, and S. He, "Community detection in social network with pairwisely constrained symmetric non-negative matrix factorization," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2015, pp. 541–546.

[24] X. Liu, W. Wang, D. He, P. Jiao, D. Jin, and C. V. Cannistraci, "Semi-supervised community detection based on non-negative matrix factorization with node popularity," *Inf. Sci.*, vol. 381, pp. 304–321, Mar. 2017.

[25] F. Ye, C. Chen, Z. Wen, Z. Zheng, W. Chen, and Y. Zhou, "Homophily preserving community detection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 8, pp. 2903–2915, Aug. 2020.

[26] H. Zhang, T. Zhao, I. King, and M. R. Lyu, "Modeling the homophily effect between links and communities for overlapping community detection," in *Proc. 25th Int. Joint Conf. Artif. Intell. (IJCAI)*, Jul. 2016, pp. 3938–3944.

[27] F. Ye, S. Li, Z. Lin, C. Chen, and Z. Zheng, "Adaptive affinity learning for accurate community detection," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2018, pp. 1374–1379.

[28] H. Zhang, I. King, and M. R. Lyu, "Incorporating implicit link preference into overlapping community detection," in *Proc. 29th AAAI Conf. Artif. Intell.*, Jan. 2015, pp. 396–402.

[29] J. Tang, Y. Chang, C. Aggarwal, and H. Liu, "A survey of signed network mining in social media," *ACM Comput. Surv.*, vol. 49, no. 3, pp. 1–37, Dec. 2016.

[30] C. Ding, T. Li, and M. I. Jordan, "Convex and semi-nonnegative matrix factorizations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 1, pp. 45–55, Jan. 2010.

[31] C. Yan, H.-M. Cheng, X. Liu, and Z.-Y. Zhang, "Joint non-negative matrix factorization for community structures detection in signed networks," 2018, *arXiv:1807.08281*. [Online]. Available: http://arxiv.org/abs/1807.08281

[32] L. Ou-Yang, D.-Q. Dai, and X.-F. Zhang, "Detecting protein complexes from signed protein-protein interaction networks," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 12, no. 6, pp. 1333–1344, Nov. 2015.

[33] C. Yan and Z. Chang, "Modularized convex nonnegative matrix factorization for community detection in signed and unsigned networks," *Phys. A, Stat. Mech. Appl.*, vol. 539, Feb. 2020, Art. no. 122904.

[34] A. Salah, M. Ailem, and M. Nadif, "A way to boost semi-NMF for document clustering," in *Proc. ACM Conf. Inf. Knowl. Manage. (CIKM)*, Nov. 2017, pp. 2275–2278.

[35] K. Allab, L. Labiod, and M. Nadif, "A semi-NMF-PCA unified framework for data clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 1, pp. 2–16, Jan. 2017.

[36] Z. Li, J. Chen, Y. Fu, G. Hu, Z. Pan, and L. Zhang, "Community detection based on regularized semi-nonnegative matrix tri-factorization in signed networks," *Mobile Netw. Appl.*, vol. 23, pp. 71–79, May 2018.

[37] J.-Y. Shi, K.-T. Mao, H. Yu, and S.-M. Yiu, "Detecting drug communities and predicting comprehensive drug–drug interactions via balance regularized semi-nonnegative matrix factorization," *J. Cheminformat.*, vol. 11, no. 1, pp. 1–16, Apr. 2019.

[38] C. B. He *et al.*, "Similarity preserving overlapping community detection in signed networks," *Future Gener. Comput. Syst.*, vol. 116, pp. 275–290, Mar. 2021.

[39] P. Chunaev, "Community detection in node-attributed social networks: A survey," *Comput. Sci. Rev.*, vol. 37, Aug. 2020, Art. no. 100286.

[40] S. Chang, G.-J. Qi, C. C. Aggarwal, J. Zhou, M. Wang, and T. S. Huang, "Factorized similarity learning in networks," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Dec. 2014, pp. 60–69.

[41] Z. Huang, Y. Ye, X. Li, F. Liu, and H. Chen, "Joint weighted nonnegative matrix factorization for mining attributed graphs," in *Proc. 21st Pacific–Asia Conf. Knowl. Discovery Data Mining (PAKDD)*, May 2017, pp. 368–380.

[42] Y. Pei, N. Chakraborty, and K. Sycara, "Nonnegative matrix tri-factorization with graph regularization for community detection in social networks," in *Proc. 24th Int. Joint Conf. Artif. Intell. (IJCAI)*, Jul. 2015, pp. 2083–2089.

[43] T. Guo, S. Pan, X. Zhu, and C. Zhang, "CFOND: Consensus factorization for co-clustering networked data," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 4, pp. 706–719, Apr. 2019.

[44] X. Wang, D. Jin, X. Cao, L. Yang, and W. Zhang, "Semantic community identification in large attribute networks," in *Proc. 30th AAAI Conf. Artif. Intell.*, 2016, pp. 265–271.

[45] M. Qin, D. Jin, K. Lei, B. Gabrys, and K. Musial-Gabrys, "Adaptive community detection incorporating topology and content in social networks," *Knowl. Based Syst.*, vol. 161, pp. 342–356, Dec. 2018.

[46] R. Li, F. Ye, S. Xie, C. Chen, and Z. Zheng, "Digging into it: Community detection via hidden attributes analysis," *Neurocomputing*, vol. 331, pp. 97–107, Feb. 2019.

[47] D. Jin, Z.-Y. Liu, R.-F. He, X. Wang, and D.-X. He, "A robust and strong explanation community detection method for attributed networks," *Chin. J. Comput.*, vol. 41, no. 7, pp. 1476–1489, Jul. 2018.

[48] E. E. Papalexakis, L. Akoglu, and D. Ience, "Do more views of a graph help? Community detection and clustering in multi-graphs," in *Proc. 16th Int. Conf. Inf. Fusion (ICIF)*, Jul. 2013, pp. 899–905.

[49] J. Kim and J.-G. Lee, "Community detection in multi-layer graphs: A survey," *ACM SIGMOD Rec.*, vol. 44, no. 3, pp. 37–48, 2015.

[50] V. Gligorijevic, Y. Panagakis, and S. Zafeiriou, "Fusion and community detection in multi-layer graphs," in *Proc. 23rd Int. Conf. Pattern Recognit. (ICPR)*, Dec. 2016, pp. 1327–1332.

[51] V. Gligorijevic, Y. Panagakis, and S. P. Zafeiriou, "Non-negative matrix factorizations for multiplex network analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 4, pp. 928–940, Apr. 2019.

[52] X. Dong, P. Frossard, P. Vandergheynst, and N. Nefedov, "Clustering on multi-layer graphs via subspace analysis on Grassmann manifolds," *IEEE Trans. Signal Process.*, vol. 62, no. 4, pp. 905–918, Feb. 2014.

[53] T. M. G. Tennakoon, K. Luong, W. Mohotti, S. Chakravarthy, and R. Nayak, "Multi-type relational data clustering for community detection by exploiting content and structure information in social networks," in *Proc. 16th Pacific Rim Int. Conf. Artif. Intell. (PRICAI)*, Aug. 2019, pp. 541–554.

[54] Y. Ma, X. Hu, T. He, and X. Jiang, "Clustering and integrating of heterogeneous microbiome data by joint symmetric nonnegative matrix factorization with Laplacian regularization," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 17, no. 3, pp. 788–795, May/Jun. 2020.

[55] X. Ma, D. Dong, and Q. Wang, "Community detection in multi-layer networks using joint nonnegative matrix factorization," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 2, pp. 273–286, Feb. 2019.

[56] G. Rossetti and R. Cazabet, "Community discovery in dynamic networks: A survey," *ACM Comput. Surv.*, vol. 51, no. 2, pp. 1–37, Jun. 2018.

[57] X. Ma and D. Dong, "Evolutionary nonnegative matrix factorization algorithms for community detection in dynamic networks," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 5, pp. 1045–1058, May 2017.

[58] X. Ma, D. Li, S. Tan, and Z. Huang, "Detecting evolving communities in dynamic networks using graph regularized evolutionary nonnegative matrix factorization," *Phys. A, Stat. Mech. Appl.*, vol. 530, Sep. 2019, Art. no. 121279.

[59] X. Ma, B. Zhang, C. Ma, and Z. Ma, "Co-regularized nonnegative matrix factorization for evolving community detection in dynamic networks," *Inf. Sci.*, vol. 528, pp. 265–279, Aug. 2020.

[60] W. Yu, W. Wang, P. Jiao, and X. Li, "Evolutionary clustering via graph regularized nonnegative matrix factorization for exploring temporal networks," *Knowl.-Based Syst.*, vol. 167, pp. 1–10, Mar. 2019.

[61] S. Wang, G. Li, G. Hu, H. Wei, Y. Pan, and Z. Pan, "Community detection in dynamic networks using constraint non-negative matrix factorization," *Intell. Data Anal.*, vol. 24, no. 1, pp. 141–161, Feb. 2020.

[62] W. Wang, P. Jiao, D. He, D. Jin, L. Pan, and B. Gabrys, "Autonomous overlapping community detection in temporal networks: A dynamic Bayesian nonnegative matrix factorization approach," *Knowl.-Based Syst.*, vol. 110, pp. 121–134, Oct. 2016.

[63] R. Márquez, R. Weber, and A. C. P. L. F. de Carvalho, "A non-negative matrix factorization approach to update communities in temporal networks using node features," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2019, pp. 728–732.

[64] P. Jiao, W. Yu, W. Wang, X. Li, and Y. Sun, "Exploring temporal community structure and constant evolutionary pattern hiding in dynamic networks," *Neurocomputing*, vol. 314, no. 7, pp. 224–233, Nov. 2018.

[65] S. Mankad and G. Michailidis, "Structural and functional discovery in dynamic networks with non-negative matrix factorization," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 88, Oct. 2013, Art. no. 042812.

[66] P. Jiao, W. Wang, and D. Jin, "Constrained common cluster based model for community detection in temporal and multiplex networks," *Neurocomputing*, vol. 275, pp. 768–780, Jan. 2018.

[67] X. Ma, P. Sun, and Y. Wang, "Graph regularized nonnegative matrix factorization for temporal link prediction in dynamic networks," *Phys. A, Stat. Mech. Appl.*, vol. 496, pp. 121–136, Apr. 2018.

[68] X. Ma, P. Sun, and G. Qin, "Nonnegative matrix factorization algorithms for link prediction in temporal networks using graph communicability," *Pattern Recognit.*, vol. 71, pp. 361–374, Nov. 2017.

[69] W. Yu, W. Wang, P. Jiao, H. Wu, Y. Sun, and M. Tang, "Modeling the local and global evolution pattern of community structures for dynamic networks analysis," *IEEE Access*, vol. 7, pp. 71350–71360, 2019.

[70] A. P. Appel, R. L. F. Cunha, C. C. Aggarwal, and M. M. Terakado, "Temporally evolving community detection and prediction in content-centric networks," in *Proc. 18th Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases (ECMLKDD)*, Sep. 2018, pp. 3–18.

[71] K. Berahmand and A. Bouyer, "LP-LPA: A link influence-based label propagation algorithm for discovering community structures in networks," *Int. J. Mod. Phys. B*, vol. 32, no. 6, Mar. 2018, Art. no. 1850062.

[72] K. Berahmand, A. Bouyer, and M. Vasighi, "Community detection in complex networks by detecting and expanding core nodes through extended local similarity of nodes," *IEEE Trans. Computat. Social Syst.*, vol. 5, no. 4, pp. 1021–1033, Dec. 2018.

[73] J. Yang and J. Leskovec, "Overlapping community detection at scale: A nonnegative matrix factorization approach," in *Proc. 6th ACM Int. Conf. Web Search Data Mining (WSDM)*, 2013, pp. 587–596.

[74] C. J. Hsieh and I. S. Dhillon, "Fast coordinate descent methods with variable selection for non-negative matrix factorization," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2011, pp. 1064–1072.

[75] R. Du, D. Kuang, B. Drake, and H. Park, "Hierarchical community detection via rank-2 symmetric nonnegative matrix factorization," *Comput. Social Netw.*, vol. 4, Sep. 2017, Art. no. 7.

[76] D. Kuang and H. Park, "Fast rank-2 nonnegative matrix factorization for hierarchical document clustering," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2013, pp. 739–747.

[77] A. Vandaele, N. Gillis, Q. Lei, K. Zhong, and I. Dhillon, "Coordinate descent methods for symmetric nonnegative matrix factorization," May 2016, *arXiv:1509.01404*. [Online]. Available: http://arxiv.org/abs/1509.01404

[78] M. Sun and H. Van Hamme, "Large scale graph regularized non-negative matrix factorization with ℓ_1 normalization based on Kullback–Leibler divergence," *IEEE Trans. Signal Process.*, vol. 60, no. 7, pp. 3876–3880, Jul. 2012.

[79] F. Liu, X. Yang, N. Guan, and X. Yi, "Online graph regularized non-negative matrix factorization for large-scale datasets," *Neurocomputing*, vol. 204, pp. 162–171, Sep. 2016.

[80] C. He, X. Fei, H. Li, Y. Tang, H. Liu, and S. Liu, "Improving NMF-based community discovery using distributed robust nonnegative matrix factorization with SimRank similarity measure," *J. Supercomput.*, vol. 74, no. 10, pp. 5601–5624, Oct. 2018.

[81] C. He, H. Li, X. Fei, A. Yang, Y. Tang, and J. Zhu, "A topic community-based method for friend recommendation in large-scale online social networks," *Concurrency Comput., Pract. Exper.*, vol. 29, no. 6, Jul. 2017, Art. no. e3924.

[82] J. Yin, L. Gao, and Z. Zhang, "Scalable nonnegative matrix factorization with block-wise updates," in *Proc. 18th Eur. Conf. Mach. Learn. Knowl. Discovery Databases (ECMLKDD)*, Sep. 2014, pp. 337–352.

[83] A. Gittens *et al.*, "Matrix factorizations at scale: A comparison of scientific data analytics in spark and C+MPI using three case studies," in *Proc. 4th Int. Conf. Big Data (Big Data)*, Dec. 2016, pp. 204–213.

[84] H. Li, K. Li, J. Peng, J. Hu, and K. Li, "An efficient parallelization approach for large-scale sparse non-negative matrix factorization using kullback-leibler divergence on multi-GPU," in *Proc. IEEE Int. Symp. Parallel Distrib. Process. Appl. IEEE Int. Conf. Ubiquitous Comput. Commun. (ISPA/IUCC)*, Dec. 2017, pp. 511–518.

[85] C. Boutsidis and E. Gallopoulos, "SVD based initialization: A head start for nonnegative matrix factorization," *Pattern Recognit.*, vol. 41, no. 4, pp. 1350–1362, Apr. 2008.

[86] H. Qiao, "New SVD based initialization strategy for non-negative matrix factorization," *Pattern Recognit. Lett.*, vol. 63, pp. 71–77, Oct. 2015.

[87] S. M. Atif, S. Qazi, and N. Gillis, "Improved SVD-based initialization for nonnegative matrix factorization using low-rank correction," *Pattern Recognit. Lett.*, vol. 122, pp. 53–59, May 2019.

[88] S. Wild, J. Curry, and A. Dougherty, "Improving non-negative matrix factorizations through structured initialization," *Pattern Recognit.*, vol. 37, no. 11, pp. 2217–2232, Nov. 2004.

[89] Y. Xue, C. S. Tong, Y. Chen, and W. S. Chen, "Clustering-based initialization for non-negative matrix factorization," *Appl. Math. Comput.*, vol. 205, no. 2, pp. 525–536, Nov. 2008.

[90] M. Rezaei, R. Boostani, and M. Rezaei, "An efficient initialization method for nonnegative matrix factorization," *J. Appl. Sci.*, vol. 11, no. 2, pp. 354–359, 2011.

[91] G. Casalino, N. D. Buono, and C. Mencar, "Subtractive clustering for seeding non-negative matrix factorizations," *Inf. Sci.*, vol. 257, pp. 369–387, Feb. 2014.

[92] A. Janecek and Y. Tan, "Using population based algorithms for initializing nonnegative matrix factorization," in *Proc. 2nd Int. Conf. Adv. Swarm Intell. (ICSI)*, Jun. 2011, pp. 307–316.

[93] X. Wang, P. Cui, J. Wang, J. Pei, W. Zhu, and S. Yang, "Community preserving network embedding," in *Proc. 31st AAAI Conf. Artif. Intell. (AAAI)*, Feb. 2017, pp. 203–209.

[94] M. Zhang and Z. Zhou, "Structural deep nonnegative matrix factorization for community detection," *Appl. Soft Comput.*, vol. 97, Dec. 2020, Art. no. 106846.

[95] Z. Yang, T. Hao, O. Dikmen, X. Chen, and E. Oja, "Clustering by non-negative matrix factorization using graph randomwalk," in *Proc. 25th Int. Conf. Neural Inf. Process. Syst. (NIPS)*, Dec. 2012, pp. 1079–1087.

[96] X. Tang *et al.*, "Link community detection by non-negative matrix factorization with multi-step similarities," *Mod. Phys. Lett. B*, vol. 30, no. 32, Nov. 2016, Art. no. 1650370.

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

[98] W. Li, J. Xie, M. Xin, and J. Mo, "An overlapping network community partition algorithm based on semi-supervised matrix factorization and random walk," *Expert Syst. Appl.*, vol. 91, pp. 277–285, Jan. 2018.

[99] Q. Lin, Y. Lin, Q. Yu, and X. Ma, "Clustering of cancer attributed networks via integration of graph embedding and matrix factorization," *IEEE Access*, vol. 8, pp. 197463–197472, 2020.

[100] A. Grover and J. Leskovec, "Node2vec: Scalable feature learning for networks," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2016, pp. 855–864.

[101] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2014, pp. 701–710.

[102] X. Huang, J. Li, and X. Hu, "Label informed attributed network embedding," in *Proc. 10th ACM Int. Conf. Web Search Data Mining (WSDM)*, Feb. 2017, pp. 731–739.

[103] H.-J. Li, L. Wang, Y. Zhang, and M. Perc, "Optimization of identifiability for efficient community detection," *New J. Phys.*, vol. 22, no. 6, Jun. 2020, Art. no. 063035.

[104] M. Moørup and L. K. Hansen, "Automatic relevance determination for multi-way models," *J. Chemometrics*, vol. 23, nos. 7–8, pp. 352–363, Aug. 2009.

[105] V. Y. F. Tan and C. Févotte, "Automatic relevance determination in nonnegative matrix factorization with the β-divergence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1592–1605, Jul. 2013.

[106] X. Shi, H. Lu, and G. Jia, "Adaptive overlapping community detection with Bayesian nonnegative matrix factorization," in *Proc. 22nd Int. Conf. Database Syst. Adv. Appl. (DSFAA)*, Mar. 2017, pp. 339–353.

[107] H. Jin, W. Yu, and S. Li, "Graph regularized nonnegative matrix tri-factorization for overlapping community detection," *Phys. A, Stat. Mech. Appl.*, vol. 515, pp. 376–387, Feb. 2019.

[108] S. Wu, A. Joseph, A. S. Hammonds, S. E. Celniker, B. Yu, and E. Frise, "Stability-driven nonnegative matrix factorization to interpret spatial gene expression and build local gene networks," *Proc. Nat. Acad. Sci. USA*, vol. 113, no. 16, pp. 4290–4295, Apr. 2016.

[109] Z.-Y. Zhang, Y. Wang, and Y.-Y. Ahn, "Overlapping community detection in complex networks using symmetric binary matrix factorization," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 87, no. 6. Jun. 2013, Art. no. 062803.

[110] F. Ye, C. Chen, Z. Zheng, R.-H. Li, and J. X. Yu, "Discrete overlapping community detection with pseudo supervision," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2019, pp. 708–717.

[111] D. Kong, C. Ding, and H. Huang, "Robust nonnegative matrix factorization using L21-norm," in *Proc. 20th ACM Int. Conf. Inf. Knowl. Manage. (CIKM)*, 2011, pp. 673–682.

[112] C. He, Q. Zhang, Y. Tang, S. Liu, and J. Zheng, "Community detection method based on robust semi-supervised nonnegative matrix factorization," *Phys. A, Stat. Mech. Appl.*, vol. 523, pp. 279–291, Jun. 2019.

[113] M. Fakhry, P. Svaizer, and M. Omologo, "Audio source separation in reverberant environments using β-divergence-based nonnegative factorization," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 25, no. 7, pp. 1462–1476, Jul. 2017.

[114] Q. Liao, N. Guan, and Q. Zhang, "Logdet divergence based sparse non-negative matrix factorization for stable representation," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2015, pp. 871–876.

[115] C. Shi, Y. Li, J. Zhang, Y. Sun, and P. S. Yu, "A survey of heterogeneous information network analysis," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 1, pp. 17–37, Jan. 2017.

[116] H.-J. Li, Z. Wang, J. Pei, J. Cao, and Y. Shi, "Optimal estimation of low-rank factors via feature level data fusion of multiplex signal systems," *IEEE Trans. Knowl. Data Eng.*, early access, Aug. 13, 2020, doi: 10.1109/TKDE.2020.3015914.

[117] Y. Chen, L. Wang, and M. Dong, "Non-negative matrix factorization for semisupervised heterogeneous data coclustering," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1459–1474, Oct. 2010.

[118] L. Yang, X. Cao, D. He, C. Wang, X. Wang, and W. Zhang, "Modularity based community detection with deep learning," in *Proc. 25th Int. Joint Conf. Artif. Intell. (IJCAI)*, Jul. 2016, pp. 2252–2258.

[119] R. Ibrahim and D. Gleich, "Nonlinear diffusion for community detection and semi-supervised learning," in *Proc. World Wide Web Conf. (WWW)*, May 2019, pp. 739–750.

[120] M. M. Fard, T. Thonet, and E. Gaussier, "Deep k-means: Jointly clustering with k-means and learning representations," *Pattern Recognit. Lett.*, vol. 138, pp. 185–192, Oct. 2020.

[121] S. Affeldt, L. Labiod, and M. Nadif, "Spectral clustering via ensemble deep autoencoder learning (SC-EDAE)," *Pattern Recognit.*, vol. 108, Dec. 2020, Art. no. 107522.

[122] Q. Feng, L. Chen, C. L. P. Chen, and L. Guo, "Deep fuzzy clustering—A representation learning approach," *IEEE Trans. Fuzzy Syst.*, vol. 28, no. 7, pp. 1420–1433, Jul. 2020.

[123] G. Trigeorgis, K. Bousmalis, S. Zafeiriou, and B. W. Schuller, "A deep matrix factorization method for learning attribute representations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 3, pp. 417–429, Mar. 2017.

[124] C. He, H. Liu, Y. Tang, X. Fei, H. Li, and Q. Zhang, "Network embedding using deep robust nonnegative matrix factorization," *IEEE Access*, vol. 8, pp. 85441–85453, 2020.

[125] J. Li, G. Zhou, Y. Qiu, Y. Wang, Y. Zhang, and S. Xie, "Deep graph regularized non-negative matrix factorization for multi-view clustering," *Neurocomputing*, vol. 390, pp. 108–116, May 2020.

[126] F. Ye, C. Chen, and Z. Zheng, "Deep autoencoder-like nonnegative matrix factorization for community detection," in *Proc. 27th ACM Int. Conf. Inf. Knowl. Manage. (CIKM)*, Oct. 2018, pp. 1393–1402.

[127] B.-J. Sun, H. Shen, J. Gao, W. Ouyang, and X. Cheng, "A non-negative symmetric encoder-decoder approach for community detection," in *Proc. ACM Conf. Inf. Knowl. Manage. (CIKM)*, Nov. 2017, pp. 597–606.

[128] H. Zhang, H. Liu, R. Song, and F. Sun, "Nonlinear non-negative matrix factorization using deep learning," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2016, pp. 477–482.

[129] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. 5th Int. Conf. Learn. Represent. (ICLR)*, Apr. 2017, pp. 1–14.

[130] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in *Proc. 6th Int. Conf. Learn. Represent. (ICLR)*, Apr. 2018, pp. 1–12.

[131] T. N. Kipf and M. Welling, "Variational graph auto-encoders," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst. Workshop Bayesian Deep Learn. (NIPS-BDL)*, Dec. 2016, pp. 1–3.

[132] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 1, pp. 4–24, Jan. 2021.

[133] S. S. Bucak and B. Gunsel, "Incremental subspace learning via non-negative matrix factorization," *Pattern Recognit.*, vol. 42, no. 5, pp. 788–797, May 2009.

[134] M. Jakomin, Z. Bosnić, and T. Curk, "Simultaneous incremental matrix factorization for streaming recommender systems," *Expert Syst. Appl.*, vol. 160, Dec. 2020, Art. no. 113685.

[135] W. Liu *et al.*, "A three-stage method for batch-based incremental non-negative matrix factorization," *Neurocomputing*, vol. 400, pp. 150–160, Aug. 2020.

Chaobo He received the B.S., M.S., and Ph.D. degrees from South China Normal University, Guangzhou, China, in 2004, 2007, and 2014, respectively.

He is currently a Professor at Zhongkai University of Agriculture and Engineering, Guangzhou, and a Visiting Scholar at the School of Computer Science, South China Normal University. He has published over 30 papers on international journals and conferences. His research interests are machine learning and social computing.

Xiang Fei received the B.S. and Ph.D. degrees from Southeast University, Nanjing, China, in 1992 and 1999, respectively.

After graduation, he worked as a Postdoctoral Research Fellow on a number of projects, including European IST Programs and EPSRC. He is currently working as a Senior Lecturer at the School of Computing, Electronics and Mathematics, Coventry University, Coventry, U.K. His current research interests include machine learning and data mining in cyber-physical systems.

Qiwei Cheng received the B.S. degree from the School of computing science, Zhongkai University of Agriculture and Engineering, Guangzhou, China, in 2019. He is currently pursuing the M.Sc. degree with the School of Computer Science, South China Normal University, Guangzhou.

His research interests are machine learning and data mining.

Hanchao Li received the B.S. degree in mathematics from the University of Warwick, Coventry, U.K., in 2013, and the M.S. degree in computing and the Ph.D. degree in computing and data science from Coventry University, Coventry, in 2015 and 2020, respectively.

Currently, he is associating a project in online social network and working on a project in stock market analysis. He has published several journal articles and conference papers. His research interests are big data, data mining, machine learning, and any mathematics-related researches.

Zeng Hu received the B.S. and M.S. degrees from Xidian University, Xi'an, China, in 2008 and 2013, respectively, and the Ph.D. degree in information and communication engineering from South China University of Technology, Guangzhou, China, in 2018.

He is currently a Lecturer at the School of Information Science and Technology, Zhongkai University of Agriculture and Engineering, Guangzhou. His recent research interests include communication engineering and machine learning.

Yong Tang received the B.S. and M.S. degrees from Wuhan University, Wuhan, China, in 1985 and 1990, respectively, and the Ph.D. degree from the University of Science and Technology of China, Hefei, China, in 2001, all in computer science.

He is currently a Professor at the School of Computer Science, South China Normal University, Guangzhou, China. He has published over 100 papers in international journals and conferences. His research interests are big data and collaborative computing.