Homophily Preserving Community Detection

Fanghua Ye, *Student Member, IEEE*, Chuan Chen, Zhiyuan Wen, Zibin Zheng, *Senior Member, IEEE*, Wuhui Chen, and Yuren Zhou

***Abstract*— As a fundamental problem in social network analysis, community detection has recently attracted wide attention, accompanied by the output of numerous community detection methods. However, most existing methods are developed by only exploiting link topology, without taking node homophily (i.e., node similarity) into consideration. Thus, much useful information that can be utilized to improve the quality of detected communities is ignored. To overcome this limitation, we propose a new community detection approach based on nonnegative matrix factorization (NMF), namely, homophily preserving NMF (HPNMF), which models not only link topology but also node homophily of networks. As such, HPNMF is able to better reflect the inherent properties of community structure. In order to capture node homophily from scratch, we provide three similarity measurements that naturally reveal the association relationships between nodes. We further present an efficient learning algorithm with convergence guarantee to solve the proposed model. Finally, extensive experiments are conducted, and the results demonstrate that HPNMF has strong ability to outperform the state-of-the-art baseline methods.**

***Index Terms*— Community detection, graph clustering, node homophily, nonnegative matrix factorization (NMF).**

I. Introduction

MANY real-world systems can be characterized by the networked data structure, such as social networks, collaboration networks, and information networks [1]. One salient property of these networks is the presence of groups of nodes with dense connections inside the same group and sparse connections across different groups. Such groups are typically defined as *communities* [2], [3]. Exploring these communities is of significant importance to understanding the structural and functional properties of networks. In addition, finding community structure has boosted many interesting applications, e.g., friend recommendation on social networks, team formation on collaboration networks, and viral marketing on information networks [4]. The task of finding communities from networks is formally referred to as *community detection*.

In recent years, community detection has attracted much attention, and extensive approaches have been proposed [3]. These approaches can be roughly categorized into two classes: 1) the methods [5]–[8] that can automatically find optimal community structures based on certain criteria, e.g., modularity [9], minimum cut [10], normalized cut [11], to name but a few and 2) the methods [12]–[15] that require a parameter to artificially specify the number of communities to detect. Although these established approaches have already achieved good performance in some cases, it is still a big challenge to find intrinsic community structures. For instance, approaches based on classic modularity criterion may suffer from the resolution limit problem [16], i.e., they cannot identify the miniature communities of networks.

Link topology, an important network description, represents a considerable amount of latent node annotation, and thus offers a promising starting point for community detection. On this account, most existing approaches struggle to discover communities-based solely on the link topology of networks [3]. However, it is not sufficient to identify intrinsic community structures by just exploiting the link topology information. For example, on social networks, users belonging to the same community may not link with each other directly. On the other hand, on coauthorship networks, researchers belonging to different research fields may have once collaborated with each other. In addition, the link topology information may be noisy due to errors in the network collection process. These missing or unexpected links show that the link topology information is not complete for high-quality community detection. Therefore, to determine the community structures of networks precisely, more (latent) information should be taken into consideration.

To compensate the incompletion of the link topology information, some existing approaches [17], [18] propose to take node attributes into consideration. However, it is difficult to obtain node attributes in some cases. Some other approaches [19], [20] take the must-link and cannot-link information as *a priori* knowledge to guide the detection of community structures. Apparently, the priori knowledge is also difficult to obtain. In most cases, we only have the pure link topology information. Thus, to identify high-quality communities, we should take full advantage of the network structure information (e.g., extracting more latent information

Manuscript received August 3, 2018; revised March 13, 2019 and May 25, 2019; accepted August 2, 2019. Date of publication September 9, 2019; date of current version August 4, 2020. This work was supported by the National Key Research and Development Program under Grant 2016YFB1000101, by the National Natural Science Foundation of China under Grant 61722214 and Grant 11801595, by the Natural Science Foundation of Guangdong under Grant 2018A030310076, and by the CCF-Tencent Open Fund WeBank Special Funding. *(Corresponding author: Chuan Chen.)*

F. Ye, C. Chen, Z. Zheng, W. Chen, and Y. Zhou are with the School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510275, China, and also with the Guangdong Key Laboratory for Big Data Analysis and Simulation of Public Opinion, Sun Yat-sen University, Guangzhou 510275, China (e-mail: smartyfh@outlook.com; chenchuan@mail.sysu.edu.cn; zhzibin@mail.sysu.edu.cn; chenwuh@mail.sysu.edu.cn; zhouyuren@mail.sysu.edu.cn).

Z. Wen is with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong (e-mail: geek00021@gmail.com).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

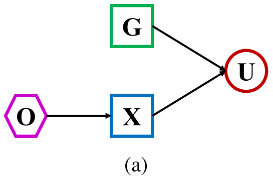
Digital Object Identifier 10.1109/TNNLS.2019.2933850

to enhance the link topology), which is undoubtedly a challenging problem.

Node homophily, another important network description, reveals the interactions between nodes, and thus should also be a valuable source of information that can be used to explain the existence of community structures in networks. According to [21], node homophily means the homogeneity (i.e., similarity) between nodes and it is a basic organization principle of networks. For example, on social networks, node homophily is the principle that communications between similar users occur at a higher rate than among dissimilar users. On coauthorship networks, node homophily gives rise to the fact that authors having lots of common articles tend to root in the same or similar research field(s). From the point of view of spreading dynamics, the direct result of node homophily is the localization of similar nodes [21], which naturally corresponds to community structures. Nevertheless, node homophily is seldom considered in the existing community detection approaches. For unannotated networks, node homophily can be extracted via deeply exploring the network structure and it is a good enhancement of the link topology information.[1] By learning node homophily, the effects of missing or unexpected links can be reduced. In short, node homophily implies the similarity between nodes, and it has significant impacts on the formation of community structures.

Therefore, the integration of the link topology information and the node homophily information provides a great potential for high-quality community detection. However, there are two challenges to overcome. First, it is challenging to precisely extract the node homophily information that can capture latent properties of communities. Second, it is technically challenging to integrate link topology and node homophily effectively. This is because when identifying communities via modeling link topology, the target is to maximize the link density within communities, while via modeling node homophily, the target is to maximize the node similarities within communities. Thus, we need to fuse the two targets into one and then optimize the two perspectives at the same time (i.e., to detect communities with both high link density and high node similarities).

Motivated by the above-mentioned analyses, we propose a novel node homophily preserving nonnegative matrix factorization (HPNMF) approach to explicitly model the effects of both link topology and node homophily under the popular nonnegative matrix factorization (NMF) [22] framework. NMF has also been employed for community detection in [12], [13], [23], and [24], where only the link topology information is exploited. By optimizing the HPNMF model, we obtain one community membership matrix U. Each row of U represents the strength of node–community relationships. For modeling link topology, our HPNMF model is based on the basic assumption that nodes sharing similar community memberships should be more likely to attach to each other. This requires that the link topology information could be recovered from U. For modeling node homophily, we provide three similarity measurements that can naturally reveal node homophily from scratch. The intuition for modeling node homophily is that similar nodes are more likely to have the same community memberships. This requires that the rows of U corresponding to similar nodes should be as close as possible to each other. To make our HPNMF model more effective, we further introduce a sparsity constraint on U, aiming to select the most relevant community for each node. The key difference between the existing approaches and our HPNMF model is illustrated in Fig. 1.



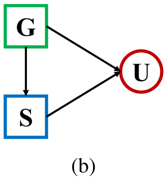


Fig. 1. Graphical models of the existing approaches and our HPNMF model. Network: G. Communities: U. Node attributes: X. Node homophily: S. External information: O. As can be seen, the existing approaches detect communities based on link topology and external information. On the other hand, our model first extracts node homophily from network structure and then integrates the link topology and node homophily information to identify communities. (a) Graphical model of existing approaches. (b) Graphical model of HPNMF.

In summary, the main contributions of this article include the following.

1) A novel community detection model, namely, HPNMF, is proposed to model the link topology and node homophily of networks simultaneously. As such, HPNMF can better characterize the properties of community structures.
2) An efficient learning algorithm with convergence guarantee is proposed to optimize the HPNMF model.
3) Extensive experiments are conducted to demonstrate that HPNMF can identify communities with better quality and higher accuracy than the state-of-the-art baseline methods.

The rest of this article is organized as follows. In Section II, we give a brief review of related work. Next, we formulate the community detection problem in Section III. The proposed HPNMF model is then detailed in Section IV, followed by the description of the learning algorithm in Section V. In Section VI, extensive experimental results to demonstrate the efficiency and effectiveness of our HPNMF model are reported, and this article is finally concluded in Section VII.

II. Related Work

Due to the ubiquity of community structures in real-world networks [2], the task of identifying communities from networks has obtained enormous attention from different research fields ranging from computer science to mathematics and physics in the past several decades [3]. This issue is formally referred to as community detection, which has been widely recognized as a fundamental problem in network analysis. Community detection has also boosted many practical applications such as friend recommendation and advertising. As a result, lots of community detection approaches have been

[1] Although link topology and node homophily are both extracted from network structure, they are in essence distinct. The object of link topology is the link, while the object of node homophily is the node. In fact, link topology and node homophily can be regarded as two views of the network structure.

proposed, such as [1], [6], [8], [13], [23], [25], [26], and so on. For detailed review of community detection methods, refer to [3].

Although so many community detection approaches have been proposed, there is no universally accepted single definition of community structures in networks. Hence, variant criteria are adopted in the literature [27]. Among them, minimum cut [10] and modularity [9] are two of the most widely adopted criteria for community detection. The minimum-cut criterion-based approaches seek to partition a network into disjoint subgraphs such that the number of cuts across these subgraphs is minimized. The modularity criterion-based approaches aim to partition a network into disjoint groups by ensuring that the number of edges within a group is significantly more than the expected number of edges. However, the minimum-cut criterion-based approaches suffer from a high computation cost and tend to find communities regardless of whether they are implicit in the network structures or not [28]. The modularity criterion-based approaches usually suffer from the resolution limit problem, i.e., they cannot detect communities smaller than a scale, which depends on the total size of networks [16].

In recent years, NMF has become a new criterion for community detection, and a variety of NMF-based community detection approaches have been proposed [12]–[14], [19], [23], [29]–[31]. As a variant of the standard matrix factorization (MF) by introducing nonnegative constraints on the factor matrices, NMF turns into an imperative tool for clustering analysis [32]–[34]. NMF has powerful interpretability derived from the nonnegative constraints and it is naturally fit for community detection. In general, NMF-based community detection approaches factorize the node adjacent matrix of a network into nonnegative low-rank factor matrices, and then the node–community relationships are extracted from the factor matrices. For instance, Wang *et al.* [13] propose a symmetric NMF model to extract communities, which factorizes the adjacent matrix into two identical low-rank matrices. Jin *et al.* [14] propose a nonnegative matrix tri-factorization approach for community detection, where a third factor matrix is introduced to capture the relationships between communities. In [23], a nonnegative symmetric encoder–decoder (NSED) approach under the NMF framework is proposed for community detection. In [29], an embedding-based approach is devised to do community detection and network embedding simultaneously, which is a modularized NMF model. In addition, a Bayesian NMF model is proposed in [31] for adaptive community detection. Two evolutionary NMF frameworks are proposed in [35] for dynamic community detection.

Although the above-mentioned community detection approaches have achieved good performance in some cases, they usually detect communities by just exploiting the link topology information of networks. Thus, their performance may degrade dramatically when the link topology information is incomplete. To address this issue, some approaches propose to involve external information or priori knowledge to enhance the link topology so as to discover better community structures [17]–[20]. However, it is difficult to obtain the compensatory information. In most cases, we only have the pure network structure information. Therefore, to detect high-quality communities, we should use the network structure information perfectly, that is, extracting more information from network structure instead of just link topology. In view of this, our HPNMF model first extracts both link topology and node homophily information from networks and then fuses them to capture the properties of community structures more precisely. Since HPNMF considers link topology and node homophily simultaneously, it is fundamentally different from the existing approaches. Moreover, HPNMF is time-efficient and will not suffer from the resolution limit problem.

III. Problem Statement

We consider an undirected and unweighted network $G = (V, E)$ with $n = |V|$ nodes and $m = |E|$ edges (i.e., links), where V and E denote the node set and edge set respectively. Network G is typically represented by a node adjacent matrix $A = (a_{ij}) \in \mathbb{R}^{n \times n}$, where $a_{ij} = 1$ if nodes i and j are connected, and $a_{ij} = 0$ otherwise. Since network G is undirected, A is a symmetric matrix. Suppose that network G consists of k communities, and let $\mathcal{C}$ denote the set of communities in G, i.e., $\mathcal{C} = \{C_i | C_i \neq \emptyset, 1 \leq i \leq k\}$, where C_i corresponds to the ith community. In this article, we focus on nonoverlapping community detection, thus the requirement $C_i \cap C_j = \emptyset$ if $i \neq j$ should be satisfied.

As aforementioned, NMF is an emerging criterion for community detection, which has great interpretability and high efficacy. In light of this, we choose NMF as the criterion in this article. The problem of community detection under the NMF framework can be formally formulated as follows.

Problem Statement 1 (Community Detection via NMF): Given an undirected and unweighted network $G = (V, E)$ with its node adjacent matrix denoted by A, the NMF-based approaches aim at partitioning G into a set of disjoint communities $\mathcal{C}$ by optimizing the following objective function:

$$\min_{U \geq 0} \mathcal{O}(A, U) \tag{1}$$

where $U = (u_{ij}) \in \mathbb{R}^{n \times k}$ is termed as the community membership matrix, and each entry u_{ij} represents the strength of node i belonging to community C_j.

The main goal of this article is to seek for a proper objective function $\mathcal{O}$ that can manifest the community structures of networks more precisely than existing methods.

IV. Node Homophily Preserving NMF

Under the NMF framework, we develop our novel HPNMF model, which takes link topology and node homophily information into consideration simultaneously. In this section, we first present how to model link topology and node homophily respectively, and then combine them into a unified model.

A. Modeling Link Topology

In a network, the link topology represents a considerable amount of latent node annotation, and thus offers a promising starting point for community analysis of the network. Since the community structure requires that nodes should be densely connected within a community and sparsely connected across communities, a basic assumption to model the link topology is that two nodes should be more likely to attach to each other if they share the same community structure. Next, we state that NMF is a natural way to match this basic assumption.

As aforementioned, given a network $G = (V, E)$, the propensities of nodes i and j belonging to community C_p are denoted as u_{ip} and u_{jp}. Consequently, the expected number of edges between nodes i and j in community C_p is $u_{ip}u_{jp}$ [24]. Summing over all communities, we obtain that the expected number of edges between nodes i and j in G is $\sum_{p=1}^{k} u_{ip}u_{jp}$. From this process of generating edges, we can see that if nodes share the same community structure (i.e., they have similar community memberships), they will have more possibilities to be linked, which naturally matches our basic assumption. Obviously, the expected number of edges between nodes i and j should be as closely consistent as possible with a_{ij}. Hence, we can formulate the following objective function:

$$\min_{U\geq 0} \mathcal{L}_{\mathrm{E}} = \sum_{i,j=1}^{n} \left(a_{ij} - \sum_{p=1}^{k} u_{ip}u_{jp} \right)^2. \tag{2}$$

It is easy to transform (2) into a matrix formulation

$$\min_{U\geq 0} \mathcal{L}_{\mathrm{E}} = \|A - UU^T\|_F^2 \tag{3}$$

where $\|\cdot\|_F$ represents the Frobenius norm.

Since we are focused on disjoint community detection, a straightforward way to determine the node–community membership for node i is to choose the community C_p whose corresponding u_{ip} has the largest value as the node–community assignment. In order to enforce this strategy, we add an l_1 norm sparsity on each row of U to produce U with many very small values and a few large values, which gives rise to the following objective function together with (3):

$$\min_{U\geq 0} \mathcal{L}_{\mathrm{E}} = \|A - UU^T\|_F^2 + \gamma \sum_{i=1}^{n} \|U_{i.}\|_1^2 \tag{4}$$

where $U_{i.}$ denotes the ith row of U, $\|U_{i.}\|_1$ represents the sparsity regularization on $U_{i.}$, and γ is a positive parameter used to make a tradeoff between the first error term and the second sparsity term. Except for the sparsity constraint, the second term can also prevent the values of some rows of U too large, which guarantees that the community membership of each node can be captured by U.

B. Modeling Node Homophily

In a network, the node homophily refers to the tendency of characteristics of different nodes to be correlated, i.e., the similarity of nodes. Since the relationships between nodes with similar characteristics tend to be stronger than the relationships between nodes having different characteristics, the node homophily ought to provide a good explanation to the existence of community structures in networks.

However, as stated in Section I, most existing approaches infer network communities based on link topology directly. Without considering node homophily, these approaches neglect much useful information that can be used to improve the quality of detected communities. For example, some links may be invalid due to mistakes in the data collection process. Obviously, trying to model these noisy links while ignoring the node homophily may result in low-quality communities. But on the other hand, by utilizing the node homophily, the impacts of these noisy links can be reduced. Therefore, it is essential to take node homophily into consideration.

In light of the above discussion, the key point of modeling node homophily is to calculate the similarity between each pair of nodes. The more similar two nodes are, the more similar their community memberships are. Recall that we are focused on undirected and unweighted networks in this work. Thus, it is challenging to calculate the similarity between nodes precisely, for the node homophily can be affected by many internal and external factors, such as the locations of nodes, the neighbors of nodes, and the actual objects of nodes. To tackle this challenge, we introduce three assumptions for modeling node homophily from scratch as follows.

1) *Assumption 1:* Connected nodes should have much more similar community memberships than unconnected ones.
2) *Assumption 2:* Connected nodes with a large number of common neighbors should have much more similar community memberships than connected nodes having few common neighbors.
3) *Assumption 3:* Unconnected nodes with a large number of common neighbors should also have similar community memberships.

Assumption 1 embodies the smoothness or consistency requirement that the label of each node tends to be the same as that of its neighbors [36], [37]. But it only focuses on whether two nodes are connected or not, which makes it unable to handle the impacts of noisy links and incomplete links. Assumptions 2 and 3 can overcome this limitation due to their consideration of the similarity between nodes' neighborhood network structures. More specifically, Assumption 2 is applicable to networks with noisy links, and Assumption 3 is useful when networks are incomplete. These three assumptions capture the first-order and second-order node homophily well, yet higher order node homophily is left out, which we leave as future work.

It is not straightforward to take full advantage of the three assumptions above. Next, we provide three approaches to calculating the similarity between each pair of nodes, which naturally match the three assumptions. We denote the similarity between nodes i and j as s_{ij}. The node similarity matrix of all the nodes is then $S = (s_{ij}) \in \mathbb{R}^{n\times n}$.

1) *Naive Similarity:* This is the simplest similarity measurement, which is defined as follows:

$$s_{ij}^{(N)} = \begin{cases} 1, & \text{if } (i,j) \in E \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

2) *Jaccard Similarity:* This is based on the well-known Jaccard index, a common index for binary variables. Let $\Gamma(i)$ be the set containing node i and its adjacent neighbors, that is,

$$\Gamma(i) = \{j \in V | (i,j) \in E\} \cup \{i\}. \tag{6}$$

Then, the Jaccard similarity between nodes i and j is defined as

$$s_{ij}^{(J)} = \frac{|\Gamma(i) \cap \Gamma(j)|}{|\Gamma(i) \cup \Gamma(j)|} \tag{7}$$

where $|\cdot|$ denotes the cardinality of a set.

3) *Hybrid Similarity:* The hybrid similarity is defined as

$$s_{ij}^{(H)} = \begin{cases} s_{ij}^{(J)} \exp\left(s_{ij}^{(J)} - \bar{s}^{(J)}\right), & \text{if } s_{ij}^{(N)} > 0 \\ s_{ij}^{(J)}, & \text{otherwise} \end{cases} \tag{8}$$

where $\bar{s}^{(J)}$ denotes the average value of the Jaccard similarity between all pairs of nodes. Clearly, this definition will enhance the similarity between those pairs of linked nodes whose Jaccard similarity is above $\bar{s}^{(J)}$ and depress the similarity between those pairs of linked nodes whose Jaccard similarity is below $\bar{s}^{(J)}$.

As can be seen, the naive similarity measurement only matches Assumption 1. The Jaccard similarity measurement matches both Assumptions 2 and 3, but it makes no distinction between them due to the lack of emphasis on the importance of link relationships between nodes. With the aids of the enhancing and depressing strategies, the hybrid similarity measurement is a natural match to all the three assumptions, which means that the hybrid similarity measurement is more robust to missing and noisy links. In the remainder of this article, we remove the superscripts of s_{ij} for brevity.

Let $U_{i.}$ and $U_{.i}$ denote the ith row and ith column of U, respectively. Based on the node similarity matrix S, we can then formulate the following objective function:

$$\begin{aligned} \min_{U \geq 0} \mathcal{L}_{\text{V}} &= \frac{1}{2} \sum_{i,j=1}^{n} s_{ij} \|U_{i.} - U_{j.}\|_2^2 \\ &= \frac{1}{2} \sum_{i,j=1}^{n} s_{ij} \sum_{p=1}^{k} (u_{ip} - u_{jp})^2 = \sum_{p=1}^{k} U_{.p}^T L U_{.p} \\ &= \sum_{p=1}^{k} (U^T L U)_{pp} = tr(U^T L U) \end{aligned} \tag{9}$$

where $tr(\cdot)$ represents the matrix trace, and L is the graph Laplacian matrix defined by $L = D - S$. Here, $D = (d_{ij}) \in \mathbb{R}^{n \times n}$ is a diagonal matrix with $d_{ii} = \sum_{j=1}^{n} s_{ij}$. We incorporate the factor $\frac{1}{2}$ into (9) because each pair of nodes is calculated twice.

By minimizing $\mathcal{L}_{\text{V}}$, we expect that if two nodes are similar, then their community memberships are also similar to each other. However, the objective function in (9) may suffer from the trivial solution problem as pointed out in [38], that is, all columns of U tend to be identical up to a scale. To address this issue, more constraints should be imposed on U. Similar to [38], we add an orthogonal constraint on U, which leads to the following new objective function:

$$\min \mathcal{L}_{\text{V}} = tr(U^T L U), \quad \text{s.t. } U \geq 0, \quad U^T U = I. \tag{10}$$

C. Unified Model

Now, we can combine the objective function of modeling link topology in (4) and that of modeling node homophily in (10) into a unified model, which leads to the overall objective function of our HPNMF model

$$\begin{aligned} \min_{U \geq 0, U^T U = I} \mathcal{L} &= \mathcal{L}_{\text{E}} + \lambda \mathcal{L}_{\text{V}} \\ &= \|A - UU^T\|_F^2 + \lambda tr(U^T L U) + \gamma \|U 1_k\|_2^2 \end{aligned} \tag{11}$$

where λ is a positive parameter used to adjust the contribution of the node homophily information, 1_k denotes an all-one column vector with k elements, and $\|\cdot\|_2$ represents the l_2 norm. The objective function $\mathcal{L}$ in (11) is nonconvex over U, as shown in Theorem 1.

Theorem 1: The objective function $\mathcal{L}$ in (11) is not convex over U.

The detailed proof is provided in Appendix A.

V. Optimization

The orthogonal constraint on U makes (11) NP-hard to optimize. Instead of optimizing (11) directly, we choose to optimize the following objective function:

$$\begin{aligned} \min_{U \geq 0} \mathcal{L} = \|A - UU^T\|_F^2 &+ \lambda tr(U^T L U) \\ &+ \gamma \|U 1_k\|_2^2 + \alpha \|U^T U - I\|_F^2 \end{aligned} \tag{12}$$

where α controls the degree of the orthogonal constraint.

A. Learning Algorithm

First, we rewrite the objective function $\mathcal{L}$ in (12) as

$$\begin{aligned} \mathcal{L} = tr(AA^T) &+ (\alpha + 1) tr(UU^T UU^T) + \lambda tr(U^T L U) \\ &+ \gamma tr(UMU^T) - 2 tr(A' UU^T) \end{aligned} \tag{13}$$

where $M = 1_k 1_k^T$ is a $k \times k$ matrix filled with ones, and $A' = A + \alpha I$.

To solve (13), we need to introduce a Lagrange multiplier matrix $\Theta = (\theta_{ij}) \in \mathbb{R}^{n \times k}$ for the nonnegative constraints on U, which leads to an equivalent objective function $\mathcal{L}_\Theta$

$$\begin{aligned} \mathcal{L}_\Theta &= \mathcal{L} - tr(\Theta U^T) \\ &= tr(AA^T) + (\alpha + 1) tr(UU^T UU^T) + \lambda tr(U^T L U) \\ &\quad + \gamma tr(UMU^T) - 2 tr(A' UU^T) - tr(\Theta U^T). \end{aligned} \tag{14}$$

The partial derivative of $\mathcal{L}_\Theta$ with respect to U is

$$\frac{\partial \mathcal{L}_\Theta}{\partial U} = 4(\alpha + 1) UU^T U - 4A'U + 2\lambda LU + 2\gamma UM - \Theta. \tag{15}$$

Setting (15) to 0, we obtain

$$\Theta = 4(\alpha + 1) UU^T U - 4A'U + 2\lambda LU + 2\gamma UM. \tag{16}$$

Following the Karush–Kuhn–Tucker (KKT) conditions [39] that $\theta_{ij} u_{ij} = 0$, we have

$$(2(\alpha + 1) UU^T U - 2A'U + \lambda LU + \gamma UM)_{ij} u_{ij} = 0. \tag{17}$$

Algorithm 1 Algorithm for the Proposed HPNMF Model

```
Input: Node adjacent matrix $A$, node similarity matrix $S$, and parameters: $k$, $\lambda$, $\gamma$, and $\alpha$;
Output: The community membership matrix $U$;
Randomly initialize $U \geq 0$;
while not converge do
    Update $U$ according to (18);
end while
return $U$;
```

This is the fixed-point equation that the solution should satisfy at convergence. By solving this equation, we derive the following iterative updating rule:

$$u_{ij} \leftarrow u_{ij}\left(1-\beta+\beta\frac{(2A'U+\lambda SU)_{ij}}{(\Phi+\lambda DU+\gamma UM)_{ij}}\right) \qquad (18)$$

where $\Phi = 2(\alpha+1)UU^TU$, and $0 < \beta \leq 1$ is a parameter which is suggested to be 0.5 by following the strategy in [40]. Clearly, this updating rule guarantees U remains nonnegative in each iteration. Note that in the derivation of (18), we have used the fact that $L = D - S$.

Based on the updating rule in (18), the proposed optimization procedure for problem (12) is then summarized in Algorithm 1.

B. Correctness Analysis

The correctness of the updating rule in (18) is illustrated in the following theorem.

Theorem 2: If the iterative updating rule in (18) converges, then the final solution U satisfies the KKT optimality condition.

See Appendix B for a detailed proof.

C. Convergence Analysis

The proof of convergence follows a similar procedure as described in [22] by leveraging the auxiliary function.

Definition 1: A function $H(u, u')$ is referred to as an auxiliary function of function $F(u)$, if the conditions

$$H(u,u') \geq F(u) \quad \text{and} \quad H(u,u) = F(u) \qquad (19)$$

are simultaneously satisfied [22].

Lemma 1: If H is an auxiliary function of F, then F is nonincreasing under the updating rule

$$u^{(t+1)} = \arg\min_{u} H(u, u^{(t)}). \qquad (20)$$

Proof of this lemma can be found in [22]. For each element u in U, let $F(u)$ denote the part of the objective function $\mathcal{L}$ in (12), which is only related to u. A specific form of the auxiliary function $H(u, u')$ for $F(u)$ is provided in Lemma 2.

Lemma 2: The function

$$\begin{aligned}H(u,u') &= F(u') + F'(u')(u-u')\\ &\quad + \frac{(2(\alpha+1)UU^TU+\lambda DU+\gamma UM)_{ij}}{\beta u'}(u-u')^2\end{aligned} \qquad (21)$$

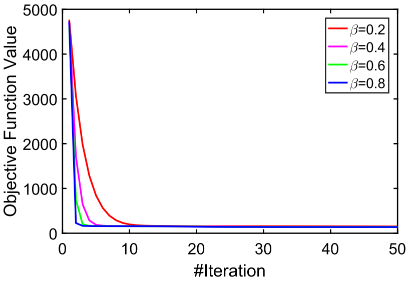


Fig. 2. Empirical convergence analysis of the updating rule in (18) on PoliticsUK by varying β from 0.2 to 0.8. As can be seen, for all the cases, the objective function value converges very fast (almost within ten iterations).

is an auxiliary function for function $F(u)$. Here, i and j are the indices of u' in U and $F'(u')$ is the first-order derivative of F with respect to u'.

Proof of this lemma is given in Appendix C. Based on Lemmas 1 and 2, we can now demonstrate the convergence of the iterative updating rule in (18).

Theorem 3: The objective function $\mathcal{L}$ in (12) is nonincreasing under the iterative updating rule in (18).

Proof of this theorem is presented in Appendix D. Here, we illustrate an empirical convergence analysis of the updating rule on the PoliticsUK data set (see Section VI for details) as shown in Fig. 2.

D. Complexity Analysis

The time complexity of Algorithm 1 is analyzed in the following theorem.

Theorem 4: Let t be the number of iterations of Algorithm 1 to achieve convergence. Then, the overall time complexity of Algorithm 1 is of order $O(t(n^2k + nk^2))$.

Proof: The main time cost of Algorithm 1 lies in the update of U. In the while loop, the update of U takes $O(n^2k + nk^2)$ time in each iteration. Therefore, the total time complexity of Algorithm 1 is of order $O(t(n^2k + nk^2))$. □

VI. Experiments

In this section, we conduct extensive experiments to validate the performance of our proposed HPNMF model by comparing it with several representative community detection approaches on nine real-world networks. All methods are implemented in Python 3.5 and all experiments are conducted on a server with two 2.4-GHz Intel Xeon CPUs and 128-GB main memory running Ubuntu 14.04.5 (64-bit).

A. Data Sets

For evaluation, we adopt nine real-world networks with different types as benchmarks, where six of them have ground-truth communities and the other three have no ground-truth communities. The detailed statistics of these networks are summarized in Table I.

These networks are downloaded from four different websites: Network Repository[2] (Dolphins, NetScience, and WikiVote) [41], Insight Resources[3] (PoliticsIE, PoliticsUK, and

[2] http://networkrepository.com/

[3] http://mlg.ucd.ie/index.html

TABLE I

NETWORK STATISTICS. n: NUMBER OF NODES, m: NUMBER OF EDGES, AND k: NUMBER OF GROUND-TRUTH COMMUNITIES

Type	Network	n	m	k
Biological	Dolphins	62	159	-
Co-authorship	NetScience	379	914	-
Web	WikiVote	889	2914	-
Social	PoliticsIE	348	12567	7
	PoliticsUK	419	19950	5
Sports	Olympics	464	7787	28
Email	EmailEU	1005	25571	42
Citation	Cora	2708	5429	7
	Pubmed	19717	44338	3

Olympics) [42], SNAP[4] (EmailEU) [43], and LINQS[5] (Cora and Pubmed).

B. Baseline Methods

We select two classes of community detection methods for comparison, including: 1) the methods that automatically find optimal community structures based on certain criteria and 2) the methods with a parameter to specify the number of desired communities.

For networks without ground-truth communities, we compare HPNMF with five methods, which fall into the first class.

LE: LE represents the leading eigenvector method, a popular community detection algorithm proposed by Newman [5]. LE is a divisive algorithm, and each split is done by maximizing the modularity regarding the original network.

LPA: LPA finds the community structures of a network according to the label propagation algorithm [6].

Infomap: Infomap is an information theoretical method [7]. Infomap envisions the community detection problem as a coding problem and aims to find optimal partitions based on the minimum description length principle.

WalkTrap: WalkTrap is a random walk-based community detection algorithm [8]. The basic idea of WalkTrap is that short random walks tend to stay in the same community.

MaxPerm: MaxPerm is a novel community detection approach [44], which aims at optimizing the permanence of networks by considering the distribution of intercluster connections from a community to its neighboring communities.

For networks having ground-truth communities, we compare HPNMF with ten methods, which fall into the second class.

KM: KM is short for the canonical k-means clustering method [45]. For KM, each row of the node adjacent matrix A is treated as a feature vector [the same to principal component analysis (PCA) in the following].

PCA: PCA is short for principal component analysis [46], one famous linear dimensionality reduction method.

TSVD: TSVD performs linear dimensionality reduction by means of truncated singular value decomposition [47].

NNCut: NNCut is a well-known algorithm for graph clustering by optimizing the normalized cut criterion [11]. NNCut belongs to the family of spectral clustering.

NMF: NMF is employed for community detection in [12].

SNMF: SNMF is adopted for community detection in [13].

NSED: NSED is a nonnegative symmetric encoder–decoder approach proposed for community detection in [23].

GNMTF: GNMTF factorizes the node adjacent matrix into three low-rank factor matrices, where one factor matrix is used to capture the relationships between communities [14].

CPNE: CPNE is an embedding-based approach, which learns community structures and network representations at the same time [29].

IFS: IFS is a fast and scalable algorithm by simulating information flow on networks [26]. It is closely related to LPA; however, it needs a parameter to specify the starting nodes. Here, we select the k nodes with the largest degree as seed nodes.

C. Evaluation Metrics

We adopt five metrics to measure the quality of detected communities. For networks without ground-truth communities, we choose *modularity* Q and *permanence* P as the evaluation metrics. For networks having ground-truth communities, we employ *adjusted Rand index (ARI)*, *normalized mutual information (NMI)*, and *F-score* as the evaluation metrics. For all these metrics, larger values indicate better performance. Refer to [48] for the details of modularity, permanence, ARI, and NMI. The F-score is defined as

$$\text{F-score} = \sum_{C_i \in \mathcal{C}} \frac{|C_i|}{\sum_{C_j \in \mathcal{C}} |C_j|} \max_{C_j^* \in \mathcal{C}^*} F\left(C_i, C_j^*\right) \tag{22}$$

where $F(C_i, C_j^*)$ represents the F-score between C_i and C_j^*. The value of F-score lies in the range of [0, 1].

D. Community Detection Results

In the experiments, we choose the naive similarity measurement as the principle to construct the node similarity matrix S for its simplicity.

We first evaluate the performance of our HPNMF model on the three networks without ground-truth communities (i.e., Dolphins, NetScience, and WikiVote). As previously discussed, we employ modularity Q and permanence P as the evaluation metrics, and we select LE, LPA, Infomap, WalkTrap, and MaxPerm for comparison. For the five baseline methods, there is no need to specify the number of desired communities, because they can automatically find the optimal partitions based on certain criteria. However, a parameter k in our HPNMF model is used to specify the number of communities to detect. For a fair comparison, we vary k from 2 to 30 with step size 1, and the best result is selected as the optima of our HPNMF model. The results are reported in Figs. 3 and 4. As illustrated in Fig. 3, HPNMF outperforms all the five baseline methods in term of modularity. From Fig. 3, we can also see that the performance of LE, LPA, and MaxPerm is consistently much worse than HPNMF, while Infomap and WalkTrap tend to achieve comparable results with HPNMF. However, both Infomap and WalkTrap are unstable. For example, WalkTrap performs well on NetScience and WikiVote, but it is inferior to LE on Dolphins. As illustrated in Fig. 4, HPNMF outperforms all the five baseline methods

[4] https://snap.stanford.edu/index.html

[5] https://linqs.soe.ucsc.edu/data

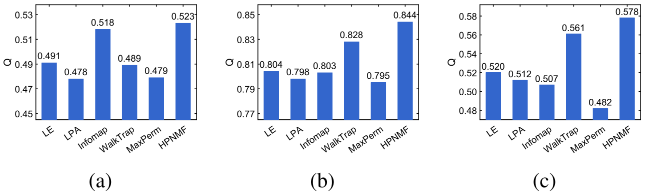


Fig. 3. Modularity on the networks without ground-truth communities. (a) Dolphins. (b) NetScience. (c) WikiVote.

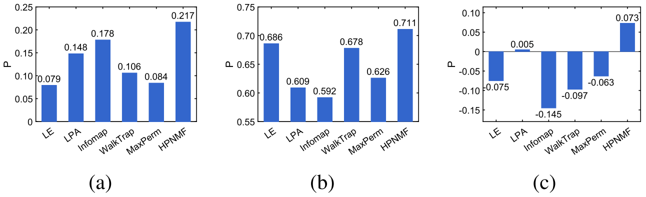


Fig. 4. Permanence on the networks without ground-truth communities. (a) Dolphins. (b) NetScience. (c) WikiVote.

in term of permanence as well. It is worthy of mentioning that MaxPerm is consistently inferior to HPNMF, although its target is to maximize the permanence when detecting communities. This is because MaxPerm is an agglomeration algorithm that needs to assign nodes to different communities heuristically, thus it is quite hard to obtain the optimal solution. These results demonstrate that when k is properly selected, HPNMF is guaranteed to detect high-quality communities.

We then evaluate the performance of HPNMF on the six networks having ground-truth communities. We compare HPNMF with KM, PCA, TSVD, NNCut, NMF, SNMF, NSED, GNMTF, CPNE, and IFS in terms of ARI, NMI, and F-score. The results are reported in Table II. This experiment aims at evaluating the accuracy of detected communities by comparing them with the ground-truth communities. Thus, the number of communities to detect for all baseline methods and our HPNMF model is set to be the ground-truth, as listed in Table I. Recall that HPNMF has two parameters λ and γ in its objective function. Parameter λ is used to adjust the relative importance of link topology and node homophily. Parameter γ is used to control the degree of sparsity regularization. In the experiment, we tune both λ and γ in the range of $\{10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}\}$. For a fair comparison, we run each method 20 times and the average results are reported.

As can be seen, HPNMF outperforms all baseline methods significantly on the six networks across different evaluation metrics, except for performing inferior to SNMF on EmailEU in term of NMI. Taking PoliticsUK as an example, we can see that HPNMF achieves over 10% performance promotion over all the three evaluation metrics (the performance promotion over ARI even reaches 30%). Table II also tells us that it is not reasonable to apply KM, PCA, and TSVD to the node adjacent matrix A by treating each row of A as a feature vector in view of their poor performance. This is because handling A in such a manner ignores much useful information, such as the interaction between nodes. From Table II, we can also see that NMF-based baseline methods tend to perform better than the other baseline methods, which shows that NMF is a powerful tool to model the link topology information. Since HPNMF outperforms all the NMF-based baseline methods, it is verified that taking node homophily into account instead of considering link topology solely is essential to improve the quality of detected communities.

The above two experiments show that our HPNMF model exploiting the link topology and node homophily information together yields better performance than all the baseline methods using only a single source of information, which justifies our motivation of taking link topology and node homophily into consideration simultaneously for community detection.

E. Similarity Measurement Comparison

As discussed in Section IV-B, we have proposed three similarity measurements (i.e., naive similarity, Jaccard similarity, and hybrid similarity) for modeling node homophily. In this experiment, we investigate the effects of the three similarity measurements. The experiment is conducted by fixing parameters λ and γ at 10^0 and 10^{-2} respectively. The results are illustrated in Fig. 5. It is observed that the Jaccard similarity measurement and hybrid similarity measurement consistently result in better performance than the naive similarity measurement. Moreover, the hybrid similarity measurement leads to the best performance. For instance, the ARI value corresponding to the hybrid similarity measurement on PoliticsIE is 0.898, which is far larger than that of the naive similarity measurement, i.e., 0.828. These results verify our statements in Section IV-B. From this experiment, we can conclude that it is necessary to differentiate different similarity measurements if we want to detect high-quality communities. It is also necessary to develop more advanced similarity measurements to reveal node homophily more accurately.

F. Parameter Sensitivity

Since there are two parameters λ and γ in the objective function of our HPNMF model, we further study how HPNMF performs when using different settings of the parameters. As stated before, we tune both λ and γ in the range of $\{10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}\}$. We report the results on PoliticsUK. Similar results can be observed on other networks. We omit them to save space.

We first evaluate the effects of λ by fixing γ at 0. The results are reported in Fig. 6(a). From Fig. 6(a), we can see that when $\lambda \leq 1$, the performance across the three evaluation metrics tends to become gradually better as λ increases, while when $\lambda > 1$, the performance tends to decreases gradually as λ increases. These results tell us that HPNMF is to some extent sensitive to parameter λ, and $\lambda = 1$ is a good choice to obtain satisfactory performance. Recall that λ controls the contribution of node homophily. The results imply that link topology and node homophily tend to be equally important for community detection. Therefore, it is essential to take node homophily into consideration.

We then evaluate the effects of γ by fixing λ at 1. The results are reported in Fig. 6(b). From Fig. 6(b), we can see that when γ takes small values, the performance of HPNMF is very stable. However, when $\gamma > 10$, the performance decreases rapidly. This is because large γ will cause the underfitting

TABLE II

PERFORMANCE COMPARISON ON THE NETWORKS WITH GROUND-TRUTH COMMUNITIES (BOLD NUMBERS REPRESENT BEST RESULTS)

Network	Metrics	KM	PCA	TSVD	NNCut	NMF	SNMF	NSED	GNMTF	CPNE	IFS	HPNMF
PoliticsIE	ARI	0.451	0.469	0.442	0.485	0.557	0.641	0.634	0.562	0.633	0.027	**0.891**
	NMI	0.623	0.619	0.602	0.661	0.673	0.760	0.767	0.647	0.708	0.049	**0.843**
	F-score	0.679	0.686	0.661	0.727	0.712	0.774	0.768	0.689	0.755	0.391	**0.935**
PoliticsUK	ARI	0.356	0.352	0.355	0.541	0.629	0.650	0.646	0.658	0.538	0.054	**0.967**
	NMI	0.534	0.534	0.535	0.631	0.726	0.724	0.736	0.694	0.552	0.056	**0.926**
	F-score	0.589	0.586	0.588	0.782	0.776	0.802	0.769	0.792	0.723	0.456	**0.976**
Olympics	ARI	0.317	0.399	0.394	0.640	0.503	0.721	0.645	0.625	0.596	0.041	**0.816**
	NMI	0.723	0.754	0.750	0.856	0.739	0.866	0.808	0.776	0.801	0.317	**0.907**
	F-score	0.590	0.632	0.627	0.794	0.662	0.811	0.744	0.718	0.703	0.256	**0.880**
EmailEU	ARI	0.056	0.082	0.086	0.071	0.264	0.455	0.469	0.452	0.482	0.033	**0.535**
	NMI	0.468	0.526	0.520	0.547	0.573	**0.692**	0.683	0.637	0.656	0.291	0.681
	F-score	0.358	0.397	0.398	0.405	0.493	0.623	0.639	0.593	0.602	0.253	**0.692**
Cora	ARI	0.015	0.019	0.020	0.006	0.182	0.231	0.175	0.110	0.099	0.129	**0.277**
	NMI	0.128	0.197	0.197	0.051	0.298	0.329	0.249	0.177	0.272	0.238	**0.352**
	F-score	0.467	0.455	0.456	0.430	0.478	0.526	0.449	0.425	0.458	0.419	**0.592**
Pubmed	ARI	-0.001	-0.002	-0.002	0.002	0.099	0.100	0.101	0.084	0.037	0.097	**0.250**
	NMI	0.027	0.034	0.035	0.043	0.151	0.158	0.144	0.101	0.142	0.093	**0.231**
	F-score	0.564	0.556	0.555	0.510	0.562	0.583	0.554	0.529	0.581	0.520	**0.650**

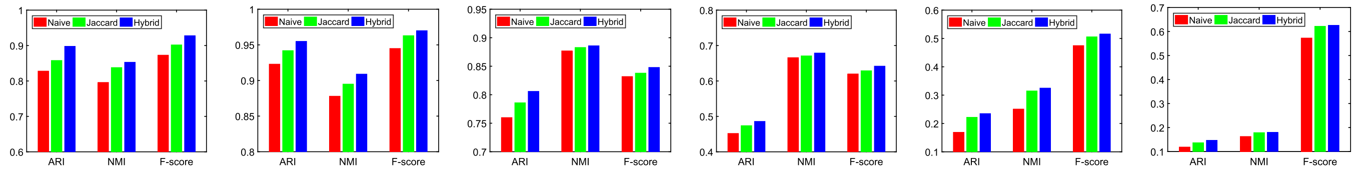


Fig. 5. Comparison of the three similarity measurements in terms of ARI, NMI, and F-score. (a) PoliticsIE. (b) PoliticsUK. (c) Olympics. (d) EmailEU. (e) Cora. (f) Pubmed.

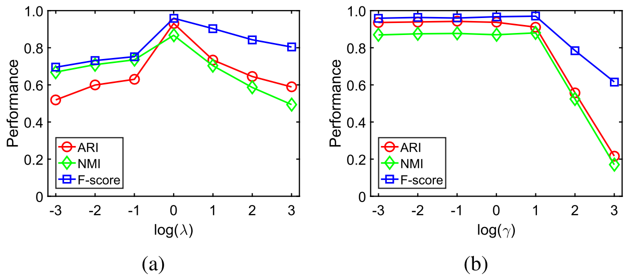


Fig. 6. Parameter sensitivity analysis on PoliticsUK. (a) Effects of λ. (b) Effects of γ.

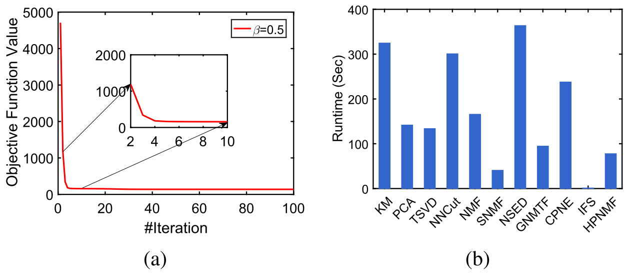


Fig. 7. Convergence analysis and runtime comparison. (a) Convergence analysis. (b) Runtime comparison.

problem. In other words, large γ will make all the elements of the community membership matrix U very small, thus it is impractical to determine community assignments based on such U. These results tell us that the sparsity coefficient γ cannot be too large.

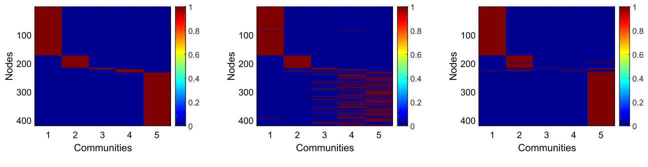
G. Convergence Study and Runtime Comparison

As aforementioned, the updating rule in (18) for minimizing the objective function $\mathcal{L}$ of our HPNMF model is essentially iterative. We have shown that $\mathcal{L}$ is nonincreasing under this updating rule in Theorem 3. In Fig. 2, we have also empirically analyzed the convergence of the updating rule on PoliticsUK by varying β from 0.2 to 0.8 with step size 0.2. In this section, we further investigate the convergence and efficiency of the updating rule on PoliticsUK when $\beta = 0.5$, which is the value we fix β at in all the experiments before. The results are depicted in Fig. 7(a). From Fig. 7(a), we can see that the updating rule is able to achieve a very rapid convergence within only a few iterations (almost less than 10).

We further report the runtime of all methods on the largest network Pubmed in Fig. 7(b). In this experiment, the iteration number is set to 200 for all NMF related methods. It can be observed from Fig. 7(b) that our HPNMF model is quite efficient. It runs much faster than most baseline methods and takes less than 100s to finish 200 iterations on Pubmed. As shown in the convergence study, HPNMF can achieve convergence within almost 10 iterations, which further guarantees the efficiency of our model, as it is reasonable to set the number of iterations to be 10 in practice.

H. Community Visualization

In this section, we aim to intuitively show the effectiveness of our HPNMF model. To this end, we visualize the communities discovered by HPNMF on PoliticsUK. For comparison,



(a) (b) (c)

Fig. 8. Community visualization on PoliticsUK. (a) Ground-truth communities. (b) Communities detected by SNMF. (c) Communities detected by our HPNMF model. The figure is best viewed in color.

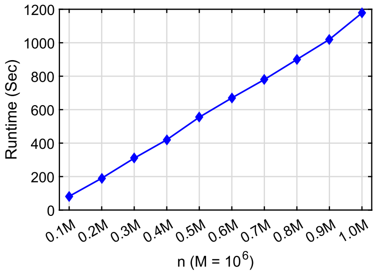


Fig. 9. Scalability testing on LFR benchmark networks via varying n.

we also visualize the ground-truth communities and the communities detected by SNMF. PoliticsUK consists of five communities, and the number of nodes in each community is 173, 43, 5, 11, and 187, respectively. For visualization, we set the element in each row of the community membership matrix U who has the largest value to 1, and all the other elements of U are set to 0. In addition, nodes with the same ground-truth community assignments are grouped together. The community visualization results are shown in Fig. 8, where the deep red color indicates the node–community relationships. As illustrated in Fig. 8, the communities discovered by our HPNMF model are very close to the ground-truth communities, even some nodes that belong to the third and fourth tiny communities are correctly clustered. While the communities detected by SNMF are very inaccurate, comparing to the ground-truth communities. Although SNMF detects the first community precisely, it fails to identify the fifth community, which is the largest one among the five ground-truth communities.

I. Scalability Testing

In this section, we further evaluate the scalability of our HPNMF model on synthetic networks. We generate ten LFR benchmark networks [49] with the number of nodes varying from one hundred thousand to one million. When generating these networks, we set the average degree to 5, the maximum degree to 30, and the other parameters of LFR are set by default. Thus, the generated networks are sparse networks. The iteration number of HPNMF is fixed at 100, and the results are illustrated in Fig. 9. As can be seen, even on the largest network (with one million nodes), HPNMF can finish running in about 20 min. The results demonstrate that HPNMF is quite efficient and can be applied to large-scale networks.

VII. Conclusion

In this article, we cope with the community detection problem. Under the NMF framework, we propose our new community detection approach HPNMF which models the link topology and node homophily of networks simultaneously. To capture node homophily from scratch, we provide three similarity measurements that can naturally reveal the association relationships between nodes. We formulate HPNMF as an optimization problem and develop an efficient learning algorithm to solve it. Extensive experiments on both real-world and synthetic networks are conducted to demonstrate the superiority of our HPNMF model.

Since we are focused on undirected and unweighted networks in this work, it would be interesting to investigate how to preserve node hompophily in directed and weighted networks. Moreover, the similarity measurements we provide can only capture the first-order and second-order node homophily, it would also be interesting to study how to embody the higher order node homophily.

Appendix A
Proof of Theorem 1

Proof: To determine whether the objective function $\mathcal{L}$ in (11) is convex or not, the main challenge lies in the first part, i.e., $\|A - UU^T\|_F^2$. Obviously, $\|A - UU^T\|_F^2$ can be treated as a multivariate polynomial of degree four, whose convexity is NP-hard to decide [50]. Therefore, it is impractical to prove its convexity. However, we can show intuitively that $\|A - UU^T\|_F^2$ is highly impossible to be convex. Suppose that $\tilde{U}$ is an optimal solution of $\|A - UU^T\|_F^2$, then we can obtain another optimal solution $\hat{U}$ by swapping any two columns of $\tilde{U}$. This is because we have $\tilde{U}\tilde{U}^T = \hat{U}\hat{U}^T$ and $\tilde{U}^T\tilde{U} = \hat{U}^T\hat{U} = I$. If $\|A - UU^T\|_F^2$ is convex over U, then according to the definition of convex function, it is straightforward that

$$\zeta\tilde{U} + (1-\zeta)\hat{U} \quad \forall\zeta \in (0, 1) \tag{23}$$

is also an optimal solution of $\|A - UU^T\|_F^2$. This phenomenon indicates that there are infinitely many optimal solutions. This is to say, there are infinitely many underlying community structures of a given network, which obviously contradicts the truth. Based on these discussion, we can conclude that $\|A - UU^T\|_F^2$ is highly impossible to be convex. Note that we also have $tr(\tilde{U}L\tilde{U}) = tr(\hat{U}L\hat{U})$ and $\|\tilde{U}1_k\|_2 = \|\hat{U}1_k\|_2$, thus following similar discussion above, we can conclude that the objective function $\mathcal{L}$ in (11) is highly impossible to be convex over U. □

Appendix B
Proof of Theorem 2

Proof: If the updating rule in (18) converges, then we have $U^{(\infty)} = U^{(t+1)} = U^{(t)} = U$, where t represents the tth iteration. That is, for each u_{ij} in U, we have

$$u_{ij} = u_{ij}\left(1 - \beta + \beta\frac{(2A'U + \lambda SU)_{ij}}{(\Phi + \lambda DU + \gamma UM)_{ij}}\right). \tag{24}$$

Then, we have

$$u_{ij} = 0, \text{ or } (\Phi - 2A'U + \lambda LU + \gamma UM)_{ij} = 0 \tag{25}$$

which results in

$$(\Phi - 2A'U + \lambda LU + \gamma UM)_{ij}u_{ij} = 0 \tag{26}$$

which is identical with (17). □

APPENDIX C
PROOF OF LEMMA 2

Proof: We prove Lemma 2 by checking whether it satisfies the two constraints defined in Definition 1. It is obvious that $H(u, u) = F(u)$, thus we only need to verify that $H(u, u') \geq F(u)$. To this end, we first rewrite $F(u)$ with the Taylor series expansion

$$F(u) = F(u') + F'(u')(u - u') + \frac{1}{2}F''(u')(u - u')^2 \qquad (27)$$

where $F'(u')$ and $F''(u')$ are the first-order and second-order derivatives of F with respect to u', respectively. Denote the indices of u' in U as i and j (i.e., u' is the element on the ith row and jth column of U), then we have

$$F'(u') = (4(\alpha+1)UU^TU - 4A'U + 2\lambda LU + 2\gamma UM)_{ij} \qquad (28)$$

and

$$\begin{aligned} F''(u') &= (-4A' + 2\lambda L + 4(\alpha + 1)UU^T)_{ii} \\ &\quad + (2\gamma M + 4(\alpha + 1)U^TU)_{jj} + 4(\alpha + 1)u'^2. \end{aligned} \qquad (29)$$

Comparing (27) with (21), we find that to verify $H(u, u') \geq F(u)$ is in essence equivalent to verify that

$$\begin{aligned} &\frac{(2(\alpha + 1)UU^TU + \lambda DU + \gamma UM)_{ij}}{\beta u'} \\ &\quad \geq (-2A' + \lambda L + 2(\alpha + 1)UU^T)_{ii} \\ &\qquad + (\gamma M + 2(\alpha + 1)U^TU)_{jj} + 2(\alpha + 1)u'^2. \end{aligned} \qquad (30)$$

It is easy to show that

$$(UU^TU)_{ij} \geq u'(UU^T)_{ii} + u'(U^TU)_{jj} - u'^3 \qquad (31)$$

and

$$(UU^TU)_{ij} \geq u'^3. \qquad (32)$$

Thus, we have

$$(UU^TU)_{ij} \geq \frac{1}{3}(u'(UU^T)_{ii} + u'(U^TU)_{jj} + u'^3). \qquad (33)$$

In addition, we have

$$(DU)_{ij} = \sum_{l=1}^{n} d_{il}u_{lj} \geq d_{ii}u' \geq (D - S)_{ii}u' = L_{ii}u' \qquad (34)$$

and

$$(UM)_{ij} = \sum_{l=1}^{k} u_{il}M_{lj} \geq u'M_{jj} = u'. \qquad (35)$$

Following (33)–(35), we can conclude that when $0 < \beta \leq (1/3)$, (30) holds and $H(u, u') \geq F(u)$.

It is hard to show rigorously that (30) still holds when $(1/3) < \beta \leq 1$. But in fact, (30) should hold when $(1/3) < \beta \leq 1$. This is because in the shrinkage processes of (31) and (32), we cast away many positive terms that can guarantee (30) holds. The empirical convergence analysis in Fig. 2 has also confirmed this inference. Therefore, we conclude that $H(u, u') \geq F(u)$ holds. □

APPENDIX D
PROOF OF THEOREM 3

Proof: The auxiliary function $H(u, u')$ in (21) is convex over u. Setting the derivative of $H(u, u')$ with respect to u to 0, we obtain

$$F'(u') + 2\frac{(2(\alpha+1)UU^TU + \lambda DU + \gamma UM))_{ij}}{\beta u'}(u^* - u') = 0. \qquad (36)$$

Furthermore, we have

$$\begin{aligned} u^* &= u'\left(1 - \frac{\beta F'(u')}{(4(\alpha + 1)UU^TU + 2\lambda DU + 2\gamma UM)_{ij}}\right) \\ &= u'\left(1 - \beta + \beta\frac{(2A'U + \lambda SU)_{ij}}{(2(\alpha + 1)UU^TU + \lambda DU + \gamma UM)_{ij}}\right) \end{aligned} \qquad (37)$$

which gives rise to the updating rule in (18). By setting $u^{(t+1)} = u^*$ and $u^{(t)} = u'$, we have the following conclusion:

$$F(u^{(t+1)}) \leq H(u^{(t+1)}, u^{(t)}) \leq H(u^{(t)}, u^{(t)}) = F(u^{(t)}).$$

Finally, following Lemma 1, we conclude that the function $F(u)$ is nonincreasing under the updating rule in (18). Since our updating rule is essentially elementwise, it is straightforward to imply that under this updating rule, the objective function $\mathcal{L}$ in (12) will be nonincreasing. □

REFERENCES

[1] Z. Zheng, F. Ye, R.-H. Li, G. Ling, and T. Jin, "Finding weighted k-truss communities in large networks," *Inf. Sci.*, vol. 417, pp. 344–360, Nov. 2017.

[2] 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.

[3] M. A. Javed, M. S. Younis, S. Latif, J. Qadir, and A. Baig, "Community detection in networks: A multidisciplinary review," *J. Netw. Comput. Appl.*, vol. 108, pp. 87–111, Apr. 2018.

[4] F. Ye, J. Liu, C. Chen, G. Ling, Z. Zheng, and Y. Zhou, "Efficient influential individuals discovery on service-oriented social networks: A community-based approach," in *Proc. ICSOC*. Cham, Switzerland: Springer, 2017, pp. 605–613.

[5] M. E. J. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Phys. Rev. E, Covering Stat. Nonlinear Biol. Soft Matter Phys.*, vol. 74, no. 3, 2006, Art. no. 036104.

[6] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," *Phys. Rev. E, Covering Stat. Nonlinear Biol. Soft Matter Phys.*, vol. 76, no. 3, 2007, Art. no. 036106.

[7] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," *Proc. Nat. Acad. Sci. USA*, vol. 105, no. 2, pp. 1118–1123, 2008.

[8] P. Pons and M. Latapy, "Computing communities in large networks using random walks," *J. Graph Algorithms Appl.*, vol. 10, no. 2, pp. 191–218, 2006.

[9] M. E. J. Newman, "Modularity and community structure in networks," *Proc. Nat. Acad. Sci. USA*, vol. 103, no. 23, pp. 8577–8582, 2006.

[10] J. Leskovec, K. J. Lang, and M. Mahoney, "Empirical comparison of algorithms for network community detection," in *Proc. 19th Int. Conf. World Wide Web*, 2010, pp. 631–640.

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

[12] S. Mankad and G. Michailidis, "Structural and functional discovery in dynamic networks with non-negative matrix factorization," *Phys. Rev. E, Covering Stat. Nonlinear Biol. Soft Matter Phys.*, vol. 88, no. 4, 2013, Art. no. 042812.

[13] 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.

[14] 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.

[15] F. Ye, C. Chen, and Z. Zheng, "Deep autoencoder-like nonnegative matrix factorization for community detection," in *Proc. CIKM*, 2018, pp. 1393–1402.

[16] S. Fortunato and M. Barthélemy, "Resolution limit in community detection," *Proc. Nat. Acad. Sci. USA*, vol. 104, no. 1, pp. 36–41, 2007.

[17] J. Yang, J. McAuley, and J. Leskovec, "Community detection in networks with node attributes," in *Proc. ICDM*, Dec. 2013, pp. 1151–1156.

[18] L. M. Smith, L. Zhu, K. Lerman, and A. G. Percus, "Partitioning networks with node attributes by compressing information flow," *ACM Trans. Knowl. Discovery From Data*, vol. 11, no. 2, p. 15, 2016.

[19] 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.

[20] L. Yang *et al.*, "Exploring the roles of cannot-link constraint in community detection via multi-variance mixed Gaussian generative model," *PLoS ONE*, vol. 12, no. 7, 2017, Art. no. e0178029.

[21] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: Homophily in social networks," *Annu. Rev. Sociol.*, vol. 27, no. 1, pp. 415–444, 2001.

[22] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Proc. NIPS*, 2000, pp. 556–562.

[23] B.-J. Sun, H. Shen, J. Gao, W. Ouyang, and X. Cheng, "A non-negative symmetric encoder-decoder approach for community detection," in *Proc. CIKM*, 2017, pp. 597–606.

[24] F. Ye, S. Li, Z. Lin, C. Chen, and Z. Zheng, "Adaptive affinity learning for accurate community detection," in *Proc. ICDM*, Nov. 2018, pp. 1374–1379.

[25] J. Shao, Z. Han, Q. Yang, and T. Zhou, "Community detection based on distance dynamics," in *Proc. KDD*, 2015, pp. 1075–1084.

[26] R. Venkatesaramani and Y. Vorobeychik, "Community detection by information flow simulation," 2018, *arXiv:1805.04920*. [Online]. Available: https://arxiv.org/abs/1805.04920

[27] S. E. Schaeffer, "Graph clustering," *Comp. Sci. Rev.*, vol. 1, no. 1, pp. 27–64, 2007.

[28] M. E. J. Newman, "Detecting community structure in networks," *Eur. Phys. J. B*, vol. 38, no. 2, pp. 321–330, 2014.

[29] X. Wang, P. Cui, J. Wang, J. Pei, W. Zhu, and S. Yang, "Community preserving network embedding," in *Proc. AAAI*, 2017, pp. 203–209.

[30] Y. Li, C. Sha, X. Huang, and Y. Zhang, "Community detection in attributed graphs: An embedding approach," in *Proc. AAAI*, 2018, pp. 338–345.

[31] X. Shi, H. Lu, and G. Jia, "Adaptive overlapping community detection with Bayesian nonnegative matrix factorization," in *Proc. DASFAA*. Cham, Switzerland: Springer, 2017, pp. 339–353.

[32] W. Wu, Y. Jia, S. Kwong, and J. Hou, "Pairwise constraint propagation-induced symmetric nonnegative matrix factorization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6348–6361, Dec. 2018.

[33] T. Liu, M. Gong, and D. Tao, "Large-cone nonnegative matrix factorization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 9, pp. 2129–2142, Sep. 2017.

[34] Z. Yang, Y. Xiang, K. Xie, and Y. Lai, "Adaptive method for nonsmooth nonnegative matrix factorization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 4, pp. 948–960, Apr. 2017.

[35] 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.

[36] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Proc. NIPS*, 2004, pp. 321–328.

[37] C. Chen, J. Xin, Y. Wang, L. Chen, and M. K. Ng, "A semisupervised classification approach for multidomain networks with domain selection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 1, pp. 269–283, Jan. 2018.

[38] Q. Gu, C. Ding, and J. Han, "On trivial solution and scale transfer problems in graph regularized NMF," in *Proc. IJCAI*, 2011, pp. 1288–1293.

[39] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[40] C. Ding, X. He, and H. D. Simon, "On the equivalence of nonnegative matrix factorization and spectral clustering," in *Proc. ICDM*, 2005, pp. 606–610.

[41] R. Rossi and N. K. Ahmed, "The network data repository with interactive graph analytics and visualization," in *Proc. AAAI*, 2015, pp. 4292–4293. [Online]. Available: http://networkrepository.com

[42] D. Greene and P. Cunningham, "Producing a unified graph representation from multiple social network views," in *Proc. 5th Annu. Web Sci. Conf.*, 2013, pp. 118–121.

[43] J. Leskovec and A. Krevl. (Jun. 2014). *SNAP Datasets: Stanford Large Network Dataset Collection*. [Online]. Available: http://snap.stanford.edu/data

[44] T. Chakraborty, S. Srinivasan, N. Ganguly, A. Mukherjee, and S. Bhowmick, "On the permanence of vertices in network communities," in *Proc. KDD*, 2014, pp. 1396–1405.

[45] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A k-means clustering algorithm," *Appl. Statist.*, vol. 28, no. 1, pp. 100–108, 1979.

[46] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics Intell. Lab. Syst.*, vol. 2, nos. 1–3, pp. 37–52, 1987.

[47] P. C. Hansen, "Truncated singular value decomposition solutions to discrete ill-posed problems with ill-determined numerical rank," *SIAM J. Sci. Statist. Comput.*, vol. 11, no. 3, pp. 503–518, 1990.

[48] T. Chakraborty, A. Dalmia, A. Mukherjee, and N. Ganguly, "Metrics for community analysis: A survey," *ACM Comput. Surv.*, vol. 50, no. 4, p. 54, 2017.

[49] A. Lancichinetti, S. Fortunato, and F. Radicchi, "Benchmark graphs for testing community detection algorithms," *Phys. Rev. E, Covering Stat. Nonlinear Biol. Soft Matter Phys.*, vol. 78, no. 4, 2008, Art. no. 046110.

[50] A. A. Ahmadi, A. Olshevsky, P. A. Parrilo, and J. N. Tsitsiklis, "NP-hardness of deciding convexity of quartic polynomials and related problems," *Math. Program.*, vol. 137, nos. 1–2, pp. 453–476, 2013.

Fanghua Ye (S'18) received the B.S. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 2016, and the M.S. degree from Sun Yat-sen University, Guangzhou, China, in 2019.

His current research interests include social network analysis, data mining, and machine learning.

Chuan Chen received the B.S. degree from Sun Yat-sen University, Guangzhou, China, in 2012, and the Ph.D. degree from Hong Kong Baptist University, Hong Kong, in 2016.

He is currently an Associate Research Fellow with the School of Data and Computer Science, Sun Yat-Sen University. His current research interests include machine learning, numerical linear algebra, and numerical optimization.

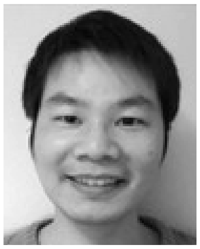
Zhiyuan Wen received the bachelor's degree in engineering from the School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China, in 2018. He is currently pursuing the Ph.D. degree with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, under the supervision of Prof. J. Cao.

His current research interests include open-domain dialogue system and text generation.

Zibin Zheng received the Ph.D. degree from the Chinese University of Hong Kong, Hong Kong, in 2011.

He is currently a Professor with the School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China, where he is also a Chairman of the Software Engineering Department. He has authored or coauthored more than 120 international journals and conference papers, including three ESI highly cited papers. According to Google Scholar, his papers have more than 7000 citations, with an H-index of 42. His current research interests include blockchain, services computing, software engineering, and financial big data.

Dr. Zheng was a recipient of several awards, including the Top 50 Influential Papers in Blockchain of 2018, the ACM SIGSOFT Distinguished Paper Award at ICSE2010, and the Best Student Paper Award at ICWS2010. He served as the General Co-Chair for BlockSys'19 and CollaborateCom'16 and the PC Co-Chair for SC2'19, ICIOT'18, and IoV'14.



Wuhui Chen received the bachelor's degree from Northeast University, Shenyang, China, in 2008, and the master's and Ph.D. degrees from the University of Aizu, Aizuwakamatsu, Japan, in 2011 and 2014, respectively.

From 2014 to 2016, he was a Research Fellow with the Japan Society for the Promotion of Science, Tokyo, Japan. From 2016 to 2017, he was a Researcher with the University of Aizu. He is currently an Associate Professor with Sun Yat-Sen University, Guangzhou, China. His current research interests include edge/cloud computing, cloud robotics, and blockchain.

Yuren Zhou received the B.Sc. degree in mathematics from Peking University, Beijing, China, in 1988, and the M.Sc. degree in mathematics and the Ph.D. degree in computer science from Wuhan University, Wuhan, China, in 1991 and 2003, respectively.

He is currently a Professor with the School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China. His current research interests include design and analysis of algorithms, evolutionary computation, and social networks.