IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 11, 2022, accepted April 11, 2022, date of publication April 18, 2022, date of current version April 27, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3168659

Community Detection in Fully-Connected Multi-layer Networks Through Joint Nonnegative Matrix Factorization

ESRAA M. AL-SHAROA[1,2], (Member, IEEE), AND SELIN AVIYENTE[2], (Senior Member, IEEE)
[1]Department of Electrical Engineering, Jordan University of Science and Technology, Irbid 22110, Jordan
[2]Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI 48824, USA

Corresponding author: Esraa M. Al-Sharoa (emalsharoa@just.edu.jo)

This work was supported in part by the Jordan University of Science and Technology and National Science Foundation under Grant CCF-2006800.

ABSTRACT Modern data analysis and processing tasks typically involve large sets of structured data. Graphs provide a powerful tool to describe the structure of such data, where the entities and the relationships between them are modeled as the nodes and edges of the graph. Traditional single layer network models are insufficient for describing the multiple entity types and modes of interaction encountered in real-world applications. Recently, multi-layer network models, which consider the different types of interactions both within and across layers, have emerged to model these systems. One of the important tools in understanding the topology of these high-dimensional networks is community detection. In this paper, a joint nonnegative matrix factorization approach is proposed to detect the community structure in multi-layer networks. The proposed approach models the multi-layer network as the union of a multiplex and bipartite network and formulates community detection as a regularized optimization problem. This optimization problem simultaneously finds the nonnegative low-rank embedding of the intra- and inter-layer adjacency matrices while minimizing the distance between the two to guarantee pair-wise similarity across embeddings. The proposed approach can detect the community structure for both homogeneous and heterogeneous multi-layer networks and is robust to noise and sparsity. The performance of the proposed approach is evaluated for both simulated and real networks and compared to state-of-the-art methods.

INDEX TERMS Multi-layer networks, community detection, intra-layer community, inter-layer community, nonnegative matrix factorization.

I. INTRODUCTION

Many real-world systems, including social and biological systems, are often represented as complex networks capturing the interactions between multiple agents [1], [2]. The different agents are represented as the nodes of the network, and the relationships among them are encoded by the edges of the network. In many real-world systems, multiple modes of interaction may exist between the different entities. These interactions may be interdependent, revealing different kinds of structure in the network. Some examples include social networks where the same people may interact in different ways such as friendship and collaborations or transportation networks where the different cities may be connected through different modes of transportation such as by air, train or bus [3], [4]. These networks can be modeled using multi-layer network models where each type of link can be separated into its own layer, thereby connecting the same set of nodes in multiple ways [5]. However, understanding the large-scale structure of multi-layer networks is made difficult by the fact that the patterns of one type of link may be similar to, uncorrelated with, or different from the patterns of another type of link. These differences from layer to layer may exist at the level of individual links, connectivity patterns among groups of nodes, or even the hidden groups themselves to which each node belongs. Therefore, finding community structure in multi-layer networks is an important problem and requires simultaneously considering the interdependence between layers and accurately defining intra-layer and inter-layer communities [6], [7].

Existing literature on community detection in multi-layer networks, e.g., [8]–[13], focuses primarily on multiplex networks, where each layer has the same set of entities of the same type. In these networks, intra-layer edges are shown

The associate editor coordinating the review of this manuscript and approving it for publication was Md. Abdur Razzaque.

explicitly while inter-layer edges are not given. Therefore, the existing methods are not directly applicable to fully connected multi-layer networks where both intra- and inter-layer edges are given explicitly. Moreover, existing approaches to multi-layer community detection mostly focus on extracting the common (or consensus) community structure across layers. As such, they are not very accurate when the layers are heterogenous.

In this paper, we address this issue by introducing a joint nonnegative matrix factorization based approach for community detection in fully-connected homogeneous and heterogeneous networks. The contributions of the proposed framework are three fold. First, the proposed approach models a multi-layer network as the union of a multiplex network and a bipartite network, where the intra-layer edges correspond to the edges of the multiplex network and the inter-layer edges correspond to the edges of the bipartite network. In this manner, the different roles that these edges play in community formation are explicitly taken into account. The respective normalized cut minimization problems for single layer and bipartite networks are then combined with the added constraint that the community assignments are consistent across the two types of networks. Second, the proposed algorithm can detect the community structure in both homogeneous and heterogeneous binary and weighted multi-layer networks efficiently and does not require all the layers to have the same number of nodes. Finally, the proposed optimization based framework is robust to noise and outliers as it employs $L_{2,1}$-norm.

This paper is organized as follows: Section II reviews the related work. Section III provides background on graph theory, nonnegative matrix factorization, community detection and asymptotical surprise metric. In Section IV, the proposed approach along with its solution and proof of convergence are presented with the additional details provided in Appendix A-D. Experiments and results are presented in Section V. Finally, the conclusions are summarized in Section VI.

II. RELATED WORK

Unlike the vast literature on community detection on single layer (monoplex) networks [14]–[16], the literature on community detection on multiplex and multilayer networks have been relatively scarce [17]. Majority of the existing work focuses on multiplex networks where there are no explicit inter-layer connectivity. Early work in this area focused on community detection by aggregating the links of all layers into a single layer, and then applying a community detection method to that single layer [18]–[21]. While these methods are computationally efficient, they are only able to identify communities that are common across all layers, and some spurious communities may emerge because of the aggregation process. State-of-the-art community detection methods for multiplex networks are built by generalizing the community detection methods from single layer.

The first class of methods is based on modularity maximization and generalizes the notion of modularity to multiple layers [11], [22], [23]. Principal Modularity Maximization (PMM) [18] extracts structural features for each layer by optimizing its modularity, and then applies PCA on concatenated matrix of structural feature matrices, to find the principal vectors, followed by K-means to perform community assignment. The main drawback of this approach is that it treats structural feature matrices of all layers equally ignoring the heterogenity across layers. On the other hand, Mucha *et al.* [11] proposed a generalized modularity metric for multiplex networks that can handle layer complementarity and an efficient algorithm, Generalized Louvain (or GenLouvain, GL), to optimize the modularity function. Despite the efficiency, the method is not designed to find consensus clustering assignment across different layers; instead, it only provides node clustering assignment for each individual layer. Moreover, this method does not handle varying strength between the nodes of different layers. In [10], a multiobjective genetic algorithm, MultiMOGA, is proposed to detect a common community structure in multidimensional networks that simultaneously maximizes the modularity on each layer and minimizes the difference between the community structure of that layer and the remaining layers. Similarly, Multi-Objective Evolutionary Algorithm based on Decomposition with Tabu Search (MOEA/D-TS), proposed in [24], detects shared communities in multiplex networks by first finding the community structure of the first layer using modularity density measure, and then formulating the problem as the maximization of modularity for each layer and the Normalized Mutual Information (NMI) between pairs of layers, simultaneously. While these papers are limited to multiplex networks, more recently a few methods have considered the extension of the modularity function and its optimization for fully-connected multi-layer networks [25], [26]. However, these methods still suffer from the resolution limit problem [27], [28].

The second class of methods relies on spectral clustering that generalizes the eigendecomposition from single to multiple Laplacian matrices representing network layers. One of the state-of-the-art spectral clustering methods for multiplex graphs is Spectral Clustering on Multi-Layer (SCML) [29]. For each network layer, SC-ML computes a subspace spanned by the principal eigenvectors of its Laplacian matrix and then, by interpreting each subspace as a point on Grassmann manifold, SC-ML merges subspaces into a consensus subspace from which the clusters are extracted. SC-ML cannot adequately handle network layers with missing or weak connections, or layers that have disconnected parts. In [30], authors propose an extension of normalized cut to multiplex networks by constructing a block Laplacian matrix with M blocks corresponding to the M layers. Standard spectral clustering is applied to this Block Laplacian matrix to obtain the community structure of the multiplex network. This method relies on the selection of a parameter

β that controls the consistency of the community structure across different layers.

The third class of methods are information diffusion-based approaches that utilize the concept of diffusion on networks to integrate network layers. One such method is Similarity Network Fusion (SNF) proposed by Wang *et al.* [31] and captures both shared and complementary information in network layers. It computes a fused matrix from the similarity matrices derived from all layers through parallel interchanging diffusion process on network layers and then applies a spectral clustering method on the fused matrix to extract communities. Another widely used method is Multiplex Infomap [9], which optimizes the map equation, and exploits the information-theoretic duality between network dimensionality reduction, and the problem of network community detection. However, for noisy networks, the diffusion process, i.e., information propagation, is not very efficient and it may result in poor clustering performance [32].

The last category of methods relies on matrix and tensor factorization and utilize collective factorization of adjacency matrices representing network layers [29], [33]–[36]. Reference [34] introduced the Linked Matrix Factorization (LMF) which fuses information from multiple network layers by factorizing each adjacency matrix into a layer-specific factor and a factor that is common to all network layers. Dong *et al.* [29], introduced the Spectral Clustering with Generalized Eigendecomposition (SC-GED) which factorizes Laplacian matrices instead of adjacency matrices. Papalexakis *et al.* [35] proposed GraphFuse, a method for clustering multi-layer networks based on sparse PARAllel FACtor (PARAFAC) decomposition with nonnegativity constraints. A similar approach has been adopted by Gauvin *et al.* [36] who used PARAFAC decomposition for time-varying networks. Cheng *et al.* [33] introduced Co-regularized Graph Clustering based on NMF (CGC-NMF), where each adjacency matrix is factorized using symmetric NMF while the Euclidean distance between their nonnegative low-dimensional representations is kept small. More recently, NMF based multiplex community detection methods using different non-negative matrix factorization models, e.g., collective symmetric NMF (CSNMF), collective projective NMF (CPNMF), collective symmetric nonnegative matrix trifactorization (CSNMTF), have been proposed [37], where first, a non-negative, low-dimensional feature representation of each network layer is found and then, the feature representation of layers are fused into a common non-negative, low-dimensional feature representation via collective factorization. In Javed *et al.* [38], SNMTF based multiplex community detection method is proposed where the objective function finds a common community structure across layers while ensuring that the common community matrix is close to all individual community matrices. In [39], authors propose a Semi-Supervised joint Nonnegative Matrix Factorization (S2-jNMF) algorithm for community detection in multiplex networks whose main goal is to detect common communities across layers. A greedy search of dense subgraphs is performed and these subgraphs are used as *a priori* information to create new adjacency matrices for each layer. The algorithm jointly decomposes these newly created adjacency matrices into a basis matrix and multiple coefficient matrices and discover the communities based on this basis matrix. This method cannot deal with fully-connected networks and heterogeneity across layers. Reference [40] use NMF for detecting communities in Multiplex Social Networks, where two different classes of approaches, Unifying and Coupling, are proposed. The first approach finds a common community structure in the networks by aggregating all layers while the second one finds mostly consistent community structures.

III. BACKGROUND

Notation: List of the main notation used in this paper is summarized in Table 1.

TABLE 1. List of notation.

Symbol	Description
$\mathbf{A}^{\alpha} \in \mathbb{R}^{n^{\alpha} \times n^{\alpha}}$	Intra-layer adjacency matrix
$\mathbf{A}^{\alpha\beta} \in \mathbb{R}^{n^{\alpha} \times n^{\beta}}$	Inter-layer adjacency matrix
$\mathbf{U}_{\alpha} \in \mathbb{R}^{n^{\alpha} \times r_{\alpha}}$	Intra-layer nonnegative embedding matrix
$\mathbf{B}_{\alpha} \in \mathbb{R}^{n^{\alpha} \times r_{\alpha\beta}}$	Inter-layer nonnegative embedding matrix
$\mathbf{S}^{\alpha\beta} \in \mathbb{R}^{r_{\alpha\beta} \times r_{\alpha\beta}}$	Nonnegative factor block matrix
$\mathcal{C}_{intra}$	Initial set of intra-layer communities
$\mathcal{C}_{inter}$	Initial set of inter-layer communities
r_{α}	Initial number of intra-layer communities
$r_{\alpha\beta}$	Initial number of inter-layer communities
μ_1, μ_2	Regularization parameters
$\|.\|_F$	Frobenious norm
$\|.\|_{2,1}$	$L_{2,1}$-norm

A. GRAPH THEORY

A multi-layer network with M layers can be defined as $\mathcal{G}_{\mathcal{M}} = (\mathcal{G}_{Intra}, \mathcal{G}_{Inter})$, where $\mathcal{G}_{Intra} = \cup_{\alpha=1}^{M} L_{\alpha}$ and $\mathcal{G}_{Inter} = \cup_{\alpha,\beta=1,\alpha\neq\beta}^{M} L_{\alpha\beta}$ represent the set of intra-layer and inter-layer graphs, respectively [5]. Each intra-layer, L_{α}, is denoted as a triplet $L_{\alpha} = (V_{\alpha}, E_{\alpha}, \mathbf{A}^{\alpha})$ where V_{α} and $E_{\alpha} \subseteq (V_{\alpha} \times V_{\alpha})$ are the set of intra-layer nodes with cardinality $|V_{\alpha}| = n^{\alpha}$ and intra-layer edges, respectively. $\mathbf{A}^{\alpha} \in \mathbb{R}^{n^{\alpha} \times n^{\alpha}}$ is the intra-layer adjacency matrix. On the other hand, inter-layer graph is denoted as a quadruplet as $L_{\alpha\beta} = (V_{\alpha}, V_{\beta}, E_{\alpha\beta}, \mathbf{A}^{\alpha\beta})$ where $E_{\alpha\beta} \subseteq (V_{\alpha} \times V_{\beta})$ with $\alpha, \beta \in \{1, 2, \ldots, M\}$ and $\alpha \neq \beta$ represents the set of inter-layer edges and $\mathbf{A}^{\alpha\beta} \in \mathbb{R}^{n^{\alpha} \times n^{\beta}}$ is the inter-layer adjacency matrix.

The adjacency matrices of the intra- and inter-layer networks, $\mathbf{A}^{\alpha} \in \mathbb{R}^{n^{\alpha} \times n^{\alpha}}$ and $\mathbf{A}^{\alpha\beta} \in \mathbb{R}^{n^{\alpha} \times n^{\beta}}$, can be used to construct a symmetric *supra-adjacency* matrix, $\mathbf{A}_{Supra} \in \mathbb{R}^{n \times n}$ where $n = \sum_{\alpha=1}^{M} n^{\alpha}$, of the multi-layer network, $\mathcal{G}_{\mathcal{M}}$, as:

$$\mathbf{A}_{Supra} = \begin{bmatrix} \mathbf{A}^{1} & \mathbf{A}^{12} & \ldots & \mathbf{A}^{1M} \\ \mathbf{A}^{21} & \mathbf{A}^{2} & \ldots & \mathbf{A}^{2M} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{A}^{M1} & \mathbf{A}^{M2} & \ldots & \mathbf{A}^{M} \end{bmatrix}. \quad (1)$$

In this paper, two types of multi-layer networks are considered: Homogeneous and heterogeneous multi-layer networks. In homogeneous multi-layer networks (HoMLNs), all layers have the same set of objects, i.e., V_α is the same as V_β for $\alpha \neq \beta$ and $n^\alpha = n^\beta$. On the other hand, in heterogeneous multi-layer networks (HeMLNs), each layer has a different set of objects, i.e., V_α is different from V_β for $\alpha \neq \beta$ and n^α may or may not be equal to n^β. For both types of networks, the *supra-adjacency* matrix can be constructed using Eq. 1.

B. NONNEGATIVE MATRIX FACTORIZATION (NMF)

Given a data matrix $\mathbf{X} \in \mathbb{R}^{n\times m}$, basic NMF factorizes $\mathbf{X}$ into the product of two nonnegative low-rank matrices as:

$$\min_{\substack{\mathbf{B}_1\in\mathbb{R}^{n\times r}\\ \mathbf{B}_2\in\mathbb{R}^{m\times r}}} \|\mathbf{X}-\mathbf{B}_1\mathbf{B}_2^\top\|_F^2, \quad s.t\ \mathbf{B}_1\geq 0,\ \mathbf{B}_2\geq 0, \tag{2}$$

where $\top$ and $\|.\|_F$ are the transpose operator and the Frobenious norm, respectively. Multiple approaches have been proposed to solve the NMF problem [41]–[43]. One of the commonly used solutions to the optimization problem in Eq. 2 is the multiplicative update rule (MUR) with the following update rules:

$$(\mathbf{B}_1)_{ij} \leftarrow \frac{(\mathbf{X}\mathbf{B}_2)_{ij}}{(\mathbf{B}_1\mathbf{B}_2^\top\mathbf{B}_2)_{ij}}(\mathbf{B}_1)_{ij}, \tag{3}$$

$$(\mathbf{B}_2)_{ij} \leftarrow \frac{(\mathbf{X}^\top\mathbf{B}_1)_{ij}}{(\mathbf{B}_2\mathbf{B}_1^\top\mathbf{B}_1)_{ij}}(\mathbf{B}_2)_{ij}. \tag{4}$$

These update rules are iterated for each variable by fixing the other one until convergence [41]–[43].

A special case of this factorization is when the input matrix is the adjacency matrix of an undirected graph, where $\mathbf{X} = \mathbf{X}^\top = \mathbf{A}$. In this special case, also known as the symmetric nonnegative matrix factorization (SymNMF), the factors $\mathbf{B}_1 = \mathbf{B}_2 = \mathbf{B}$ and the optimization problem is defined as follows:

$$\min_{\mathbf{B}\in\mathbb{R}^{n\times r}} \|\mathbf{A}-\mathbf{B}\mathbf{B}^\top\|_F^2, \quad s.t\ \mathbf{B}\geq 0, \tag{5}$$

and the update rules can be obtained by a Newton-like algorithm or an alternating nonnegative least squares (ANLS) algorithm using projected gradient methods as suggested in [44] and [45].

In addition to the aforementioned NMF approaches, [46] introduced nonnegative matrix tri-factorization (NMTF). In NMTF, an extra factor, $\mathbf{S}$, is added to the factorization to absorb the different scales of $\mathbf{X}$, $\mathbf{B}_1$ and $\mathbf{B}_2$. The purpose of the factor $\mathbf{S}$ is to provide additional degrees of freedom to maintain the accuracy of the low-rank matrix representation. NMTF problem is formulated as follows [46]:

$$\min_{\substack{\mathbf{B}_1\in\mathbb{R}^{n\times r_1},\,\mathbf{B}_2\in\mathbb{R}^{m\times r_2},\\ \mathbf{S}\in\mathbb{R}^{r_1\times r_2}}} \|\mathbf{X}-\mathbf{B}_1\mathbf{S}\mathbf{B}_2^\top\|_F^2$$
$$s.t\ \mathbf{B}_1\geq 0, \quad \mathbf{B}_2\geq 0,\ \mathbf{S}\geq 0, \tag{6}$$

and the solution of the problem can be computed using multiplicative update rules as follows:

$$(\mathbf{B}_1)_{ij} \leftarrow \frac{(\mathbf{X}\mathbf{B}_2\mathbf{S}^\top)_{ij}}{(\mathbf{B}_1\mathbf{S}\mathbf{B}_2^\top\mathbf{B}_2\mathbf{S}^\top)_{ij}}(\mathbf{B}_1)_{ij}, \tag{7}$$

$$(\mathbf{B}_2)_{ij} \leftarrow \frac{(\mathbf{X}^\top\mathbf{B}_1\mathbf{S})_{ij}}{(\mathbf{B}_2\mathbf{S}^\top\mathbf{B}_1^\top\mathbf{B}_1\mathbf{S})_{ij}}(\mathbf{B}_2)_{ij}, \tag{8}$$

$$(\mathbf{S})_{ij} \leftarrow \frac{(\mathbf{B}_1^\top\mathbf{X}\mathbf{B}_2)_{ij}}{(\mathbf{B}_1^\top\mathbf{B}_1\mathbf{S}\mathbf{B}_2^\top\mathbf{B}_2)_{ij}}(\mathbf{S})_{ij}. \tag{9}$$

One of the drawbacks of the standard NMF is its instability to noise or outliers since it uses least square error function. Consequently, a robust version of NMF is introduced in [47]. Robust NMF uses the $L_{2,1}$-norm loss function as follows:

$$\min_{\mathbf{B}_1\in\mathbb{R}^{n\times r},\,\mathbf{B}_2\in\mathbb{R}^{m\times r}} \|\mathbf{X}-\mathbf{B}_1\mathbf{B}_2^\top\|_{2,1}, \quad s.t\ \mathbf{B}_1\geq 0,\ \mathbf{B}_2\geq 0,$$
$$= \min_{\mathbf{B}_1,\,\mathbf{B}_2} \sum_{i=1}^{m} \|\mathbf{x}_i-\mathbf{B}_1\mathbf{b}_{2i}^\top\|_2, \quad s.t\ \mathbf{B}_1\geq 0,\ \mathbf{B}_2\geq 0, \tag{10}$$

where $\mathbf{B}_2 = [\mathbf{b}_{21}, \mathbf{b}_{22}, \ldots, \mathbf{b}_{2m}]^\top$ and the iterative update rules are defined as:

$$(\mathbf{B}_1)_{ij} \leftarrow \frac{(\mathbf{X}\mathbf{D}\mathbf{B}_2)_{ij}}{(\mathbf{B}_1\mathbf{B}_2^\top\mathbf{D}\mathbf{B}_2)_{ij}}(\mathbf{B}_1)_{ij}, \tag{11}$$

$$(\mathbf{B}_2)_{ij} \leftarrow \frac{(\mathbf{B}_1^\top\mathbf{X}\mathbf{D})_{ij}}{(\mathbf{B}_1^\top\mathbf{B}_1\mathbf{B}_2^\top\mathbf{D})_{ij}}(\mathbf{B}_2)_{ij}, \tag{12}$$

where $\mathbf{D} \in \mathbb{R}^{m\times m}$ represents a diagonal matrix with the ith diagonal element equal to:

$$\mathbf{D}_{ii} = \frac{1}{\sqrt{\sum_{j=1}^{n}(\mathbf{X}-\mathbf{B}_1\mathbf{B}_2^\top)_{ji}^2}} = \frac{1}{\|\mathbf{x}_i-\mathbf{B}_1\mathbf{b}_{2i}\|}. \tag{13}$$

C. COMMUNITY DETECTION

Over the past decades, numerous approaches have been proposed for graph partitioning. One of the most popular approaches is solving the minimum cut (min-cut) problem which involves optimization of different objective functions, e.g., ratio cut or normalized cut [48]. Given a static graph, $\mathcal{G} = \{V, E, \mathbf{A}\}$, where V, E and $\mathbf{A}$ are the set of nodes, edges and adjacency matrix of the graph, respectively, the min-cut problem aims to find a partition, $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_K\}$, by minimizing the following objective function:

$$\frac{1}{2}\sum_{k=1}^{K} cut(\mathcal{C}_k, \bar{\mathcal{C}}_k), \tag{14}$$

where $\bar{\mathcal{C}}_k$ is the complement of $\mathcal{C}_k$ and $cut(\mathcal{C}_k, \bar{\mathcal{C}}_k) = \sum_{i\in\mathcal{C}_k, j\in\bar{\mathcal{C}}_k} A_{ij}$ for the two disjoint sets $\mathcal{C}_k$ and $\bar{\mathcal{C}}_k$, where $\mathcal{C}_k, \bar{\mathcal{C}}_k \subset V$. While the min-cut problem is NP-hard, spectral clustering and nonnegative matrix factorization provide efficient solutions to different relaxed versions of the min-cut problem [48], [49]. In particular, spectral clustering solves the min-cut problem by retaining the orthogonality constraint on the basis matrices. On the other hand, nonnegative matrix

factorization solves the min-cut problem by retaining the nonnegativity constraint on the factor matrices. In this paper, the proposed approach takes advantage of the equivalency between spectral clustering and non-negative matrix factorization [44], [49], [50] to develop a unified community detection framework for multi-layer networks.

D. ASYMPTOTICAL SURPRISE METRIC

One of the challenges in community detection is determining the number of communities when there is no prior knowledge about the network's ground truth. Different quality metrics have been utilized to tackle this issue including eigengap criterion [30], [48], dispersion coefficent [51], [52], modularity [53] and asymptotical surprise metric [54].

In this paper, the asymptotical surprise[1] (AS) metric is adopted to determine the number of communities in the multi-layer network. For a simple graph, L_α, AS is defined as AS $= 2E_\alpha D_{KL}(q||\langle q\rangle)$, where $2E_\alpha$ is the total sum of edge weights in the adjacency matrix, $\mathbf{A}^\alpha$, $q = E_\alpha^{intra}/2E_\alpha$ is the observed ratio of the intra-cluster edge weights to the total edge weights and $\langle q\rangle = p_{intra}/p$ is the expected ratio of the total sum of intra-cluster edge weights p_{intra} to the total density of the graph, p. D_{KL} is the Kullback-Leibler divergence [55]. Specifically, the AS metric compares the distribution of the nodes and edges in the detected communities in a certain network with respect to a null model.

IV. COMMUNITY DETECTION IN MULTI-LAYER NETWORKS: JOINT NONNEGATIVE MATRIX FACTORIZATION (ML-JNMF)

A. DEFINITION OF COMMUNITIES

Let $\mathcal{G}_\mathcal{M}$ be a multi-layer network with M layers. A community $\mathcal{C}_k$ in $\mathcal{G}_\mathcal{M}$, can be defined as a tuple $\mathcal{C}_k = \{\mathcal{C}_{k,intra}, \mathcal{C}_{k,inter}\}$ which consists of a subset of nodes from one or more layers. More precisely, $\mathcal{C}_{k,intra} = (V_\alpha^{\mathcal{C}_k}, E_\alpha^{\mathcal{C}_k})$, with $V_\alpha^{\mathcal{C}_k} \subseteq V_\alpha$ and $E_\alpha^{\mathcal{C}_k} = E_\alpha \cap (V_\alpha^{\mathcal{C}_k} \times V_\alpha^{\mathcal{C}_k})$ where $\alpha \in \{1, 2, \ldots, M\}$ and $\mathcal{C}_{k,inter} = (V_\alpha^{\mathcal{C}_k}, V_\beta^{\mathcal{C}_k}, E_{\alpha\beta}^{\mathcal{C}_k})$ with $V_\alpha^{\mathcal{C}_k} \subseteq V_\alpha$, $V_\beta^{\mathcal{C}_k} \subseteq V_\beta$ and $E_{\alpha\beta}^{\mathcal{C}_k} = E_{\alpha\beta} \cap (V_\alpha^{\mathcal{C}_k} \times V_\beta^{\mathcal{C}_k})$ where $\alpha, \beta \in \{1, 2, \ldots, M\}$ and $\alpha \neq \beta$. Following the aforementioned definitions, a community, $\mathcal{C}_k$, in a multi-layer network can be either an intra-layer community or inter-layer community. Intra- and inter-layer communities can be defined as follows.

Definition 1: A community, $\mathcal{C}_k$, in a multi-layer network is called an intra-layer or within-layer community, if $\mathcal{C}_{k,inter} = \emptyset$.

Definition 2: A community, $\mathcal{C}_k$, in a multi-layer network is called an inter-layer or cross-layer community, if $\mathcal{C}_{k,inter} \neq \emptyset$.

In this paper, the objective of the proposed approach is to detect the community structure of the multi-layer network with $M = 2$, i.e., $\mathcal{G}_\mathcal{M} = \{L_1, L_2, L_{12}\}$, where $L_1 = (V_1, E_1)$, $L_2 = (V_2, E_2)$ and $L_{12} = (V_1, V_2, E_{12})$ are the intra- and inter-layer networks, respectively. In particular, we focus on detecting a set of K disjoint intra- and inter-layer communities, $\mathcal{C}_\mathcal{M} = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_K\}$.

B. PROBLEM FORMULATION

In the proposed approach, the community structure of the multi-layer network is detected by taking advantage of prior work in nonnegative low-rank approximation of static networks [44], [45] and bipartite networks which encode the relationship between different types of vertex sets [56]–[58]. In the case of a two-layer network, the inter-layer adjacency matrix $\mathbf{A}^{12}$ is similar to a bipartite network as it encodes the relationships between two sets of vertices in the two layers. However, unlike bipartite networks where there are no edges within a layer, we have intra-layer adjacency matrices, $\mathbf{A}^\alpha$. Therefore, any partitioning between layers needs to be consistent with within-layer partitioning.

Given a two-layer network, the objective function based on normalized cut can be written as:

$$\begin{aligned}\min \mathcal{J} = \min_{\substack{\mathbf{B}_\alpha \in \mathbb{R}^{n^\alpha \times r_\alpha},\ \mathbf{U}_\alpha \in \mathbb{R}^{n^\alpha \times r_\alpha},\\ \mathbf{S}^{12} \in \mathbb{R}^{r_{\alpha\beta} \times r_{\alpha\beta}}}} & \underbrace{\sum_{\alpha=1}^{2} \|\mathbf{A}^\alpha - \mathbf{U}_\alpha \mathbf{U}_\alpha^\top\|_{2,1}}_{\text{Intra-layer low-rank approximation}} \\ & + \mu_1 \underbrace{\|\mathbf{A}^{12} - \mathbf{B}_1 \mathbf{S}^{12} \mathbf{B}_2^\top\|_{2,1}}_{\text{Inter-layer low-rank approximation}} \\ & + \mu_2 \underbrace{\sum_{\alpha=1}^{2} \|\mathbf{U}_\alpha \mathbf{U}_\alpha^\top - \mathbf{B}_\alpha \mathbf{B}_\alpha^\top\|_F^2}_{\text{Pairwise Similarity}}, \\ \text{s.t } \mathbf{U}_\alpha \geq 0, \quad & \mathbf{B}_\alpha \geq 0,\ \mathbf{S}^{12} \geq 0, \end{aligned} \tag{15}$$

where $\mathbf{U}_\alpha$ and $\mathbf{B}_\alpha$ are the coefficient matrices corresponding to intra- and inter-layer adjacency matrices, respectively. $\mathbf{S}^{12}$ is the low-rank embedding of $\mathbf{A}^{12}$.

The different terms in the optimization problem in Eq. 15 achieve the following objectives:

- The first term provides nonnegative low-rank embedding of the intra-layer adjacency matrix where the embedding is used for intra-layer community community assignment.
- The second term provides nonnegative low-rank embedding of the inter-layer adjacency matrix where it is used for inter-layer community assignment.
- The last term quantifies the similarity between nonnegative embedding matrices, $\mathbf{U}_\alpha$ and $\mathbf{B}_\alpha$. This term ensures that the within-layer and across-layer community assignments are consistent with each other. Let $\mathbf{U}_\alpha \in \mathbb{R}^{n^\alpha \times r_\alpha}$ and $\mathbf{B}_\alpha \in \mathbb{R}^{n^\alpha \times r_\alpha}$ be two coefficient matrices, the similarity between the two coefficient matrices can then be computed using the following cost function:
$$P = \|K_{(\mathbf{U}_\alpha)} - K_{(\mathbf{B}_\alpha)}\|_F^2 = \|\mathbf{U}_\alpha \mathbf{U}_\alpha^\top - \mathbf{B}_\alpha \mathbf{B}_\alpha^\top\|_F^2, \tag{16}$$
where $K_{(\mathbf{B}_\alpha)}$ represents the kernel matrix of the coefficient matrix $\mathbf{B}_\alpha$. In this paper, the linear kernel function, $K_{(\mathbf{B}_\alpha)} = \mathbf{B}_\alpha \mathbf{B}_\alpha^\top$, is adopted to quantify the similarity between the coefficients matrices.
- μ_1 and μ_2 are the regularization parameters.

[1] https://github.com/CarloNicolini/communityalg

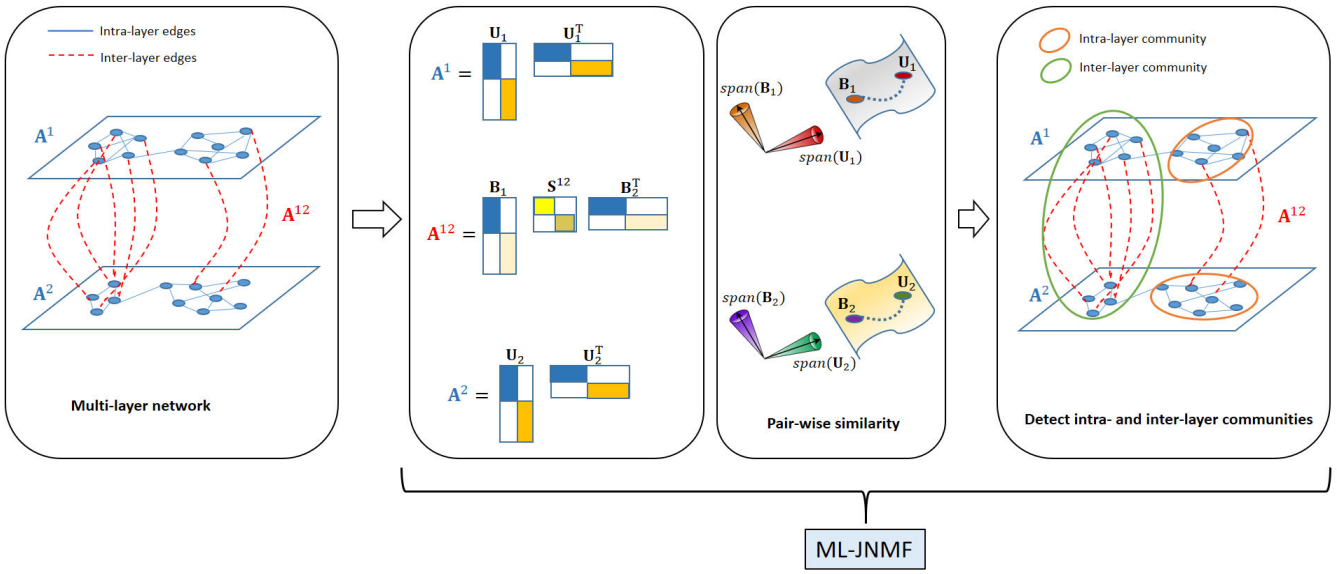

FIGURE 1. Block diagram of the proposed approach.

A block diagram illustrating the proposed approach is given in Fig. 1.

C. OPTIMIZATION ALGORITHM

Since the objective function, $\mathcal{J}$, in Eq. 15 is not convex for all variables, it is infeasible to find a global minimum for the optimization problem. However, $\mathcal{J}$ is convex in terms of each variable separately. Consequently, an iterative updating scheme can be adopted to find the local minima for the objective function. In particular, the optimization problem is divided into multiple subproblems and each variable is optimized while fixing the others. Solving for non-negative matrix factorization with $L_{2,1}$-norm was introduced in [47] using multiplicative update algorithm. In this paper, a similar update procedure will be derived to solve Eq. 15. Splitting the proposed optimization problem results in the following subproblems:

1) Fixing $\mathbf{B}_\alpha$ with $\alpha \in \{1, 2\}$ and $\mathbf{S}^{12}, \mathbf{U}_\alpha$-subproblem can be written as:

$$\min_{\mathbf{U}_\alpha} \|\mathbf{A}^\alpha - \mathbf{U}_\alpha\mathbf{U}_\alpha^\top\|_{2,1} + \mu_2\|\mathbf{U}_\alpha\mathbf{U}_\alpha^\top - \mathbf{B}_\alpha\mathbf{B}_\alpha^\top\|_F^2, \quad s.t\ \mathbf{U}_\alpha \geq 0. \tag{17}$$

2) Fixing $\mathbf{U}_\alpha$ with $\alpha \in \{1, 2\}$, $\mathbf{S}^{12}$ and $\mathbf{B}_2, \mathbf{B}_1$-subproblem can be written as:

$$\min_{\mathbf{B}_1} \mu_1\|\mathbf{A}^{12} - \mathbf{B}_1\mathbf{S}^{12}\mathbf{B}_2^\top\|_{2,1} + \mu_2\|\mathbf{U}_1\mathbf{U}_1^\top - \mathbf{B}_1\mathbf{B}_1^\top\|_F^2, \quad s.t\ \mathbf{B}_1 \geq 0. \tag{18}$$

3) Fixing $\mathbf{U}_\alpha$ with $\alpha \in \{1, 2\}$, $\mathbf{S}^{12}$ and $\mathbf{B}_1, \mathbf{B}_2$-subproblem can be written as:

$$\min_{\mathbf{B}_2} \mu_1\|\mathbf{A}^{12} - \mathbf{B}_1\mathbf{S}^{12}\mathbf{B}_2^\top\|_{2,1} + \mu_2\|\mathbf{U}_2\mathbf{U}_2^\top - \mathbf{B}_2\mathbf{B}_2^\top\|_F^2, \quad s.t\ \mathbf{B}_2 \geq 0. \tag{19}$$

4) Fixing $\mathbf{U}_\alpha$ and $\mathbf{B}_\alpha$ with $\alpha \in \{1, 2\}$, $\mathbf{S}^{12}$-subproblem can be written as:

$$\min_{\mathbf{S}^{12}} \mu_1\|\mathbf{A}^{12} - \mathbf{B}_1\mathbf{S}^{12}\mathbf{B}_2^\top\|_{2,1}, \quad s.t\ \mathbf{S}^{12} \geq 0. \tag{20}$$

The subproblems in Eqs. 17- 20 can be efficiently optimized using the following update rules:

$$(\mathbf{U}_\alpha)_{ij} \leftarrow \frac{(\mathbf{Z}^\alpha\mathbf{A}^\alpha\mathbf{U}_\alpha + \mathbf{A}^\alpha\mathbf{Z}^\alpha\mathbf{U}_\alpha + \mu_2'\mathbf{B}_\alpha\mathbf{B}_\alpha^\top\mathbf{U}_\alpha)_{ij}(\mathbf{U}_\alpha)_{ij}}{(\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{Z}^\alpha\mathbf{U}_\alpha + \mathbf{Z}^\alpha\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{U}_\alpha + \mu_2'\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{U}_\alpha)_{ij}}, \tag{21}$$

where $\mathbf{Z}_{ii}^\alpha = 1/\|\mathbf{a}_i^\alpha - \mathbf{U}_\alpha\mathbf{u}_{\alpha i}^\top\|$ and $\mu_2' = 2\mu_2$.

$$(\mathbf{B}_1)_{ij} \leftarrow \frac{(\mu_1\mathbf{A}^{12}\mathbf{Z}^{12}\mathbf{B}_2\mathbf{S}^{12\top} + \mu_2'\mathbf{U}_1\mathbf{U}_1^\top\mathbf{B}_1)_{ij}(\mathbf{B}_1)_{ij}}{(\mu_1\mathbf{B}_1\mathbf{S}^{12}\mathbf{B}_2^\top\mathbf{Z}^{12}\mathbf{B}_2\mathbf{S}^{12\top} + \mu_2'\mathbf{B}_1\mathbf{B}_1^\top\mathbf{B}_1)_{ij}}, \tag{22}$$

$$(\mathbf{B}_2)_{ij} \leftarrow \frac{(\mu_1\mathbf{Z}^{12}\mathbf{A}^{12\top}\mathbf{B}_1\mathbf{S}^{12} + \mu_2'\mathbf{U}_2\mathbf{U}_2^\top\mathbf{B}_2)_{ij}(\mathbf{B}_2)_{ij}}{(\mu_1\mathbf{Z}^{12}\mathbf{B}_2\mathbf{S}^{12\top}\mathbf{B}_1^\top\mathbf{B}_1\mathbf{S}^{12} + \mu_2'\mathbf{B}_2\mathbf{B}_2^\top\mathbf{B}_2)_{ij}}, \tag{23}$$

where $\mathbf{Z}_{ii}^{12} = 1/\|\mathbf{a}_i^{12} - \mathbf{B}_1\mathbf{S}^{12}\mathbf{b}_{2i}^\top\|$.

$$(\mathbf{S}^{12})_{ij} \leftarrow \frac{(\mathbf{B}_1^\top\mathbf{A}^{12}\mathbf{Z}^{12}\mathbf{B}_2)_{ij}}{(\mathbf{B}_1^\top\mathbf{B}_1\mathbf{S}^{12}\mathbf{B}_2^\top\mathbf{Z}^{12}\mathbf{B}_2)_{ij}}(\mathbf{S}^{12})_{ij}. \tag{24}$$

A detailed derivation of the update rules can be found in the Appendix. With the update rules in Eqs. 21- 24, the proposed algorithm is summarized in Algorithm 1.

D. DETERMINING THE QUALITY AND NUMBER OF COMMUNITIES

The quality of the community structure and the final number of communities in the multi-layer network are determined using the following steps:

- Initialize $\mathbf{U}_1 \in \mathbb{R}^{n^1 \times r_1}$ and $\mathbf{U}_2 \in \mathbb{R}^{n^2 \times r_2}$ in Step 1 of Algorithm 1 using nonnegative double singular value decomposition[2] (NNDSVD) [59] and calculate the AS metric for varying number of communities, e.g., $(2-20)$. The initial number of communities, r_1 and r_2, for each intra-layer network, is determined as the number that maximizes the AS metric for that layer. On the other hand, the initial number of communities for the inter-layer network is determined as $r_{12} = \min\{r_1, r_2\}$ and $\mathbf{B}_1 \in \mathbb{R}^{n^1 \times r_{12}}$ and $\mathbf{B}_2 \in \mathbb{R}^{n^2 \times r_{12}}$ are initialized using NNDSVD.
- After the algorithm converges, a set of intra- and inter-layer communities are detected through Steps 9-15 of Algorithm 1. In particular, the largest entry in each row of the basis matrices indicates the community assignment of each node. At the end of this step, each node is assigned to both intra- and inter-layer communities, simultaneously.
- The quality of the detected communities is calculated in Step 10 using the communitude metric of the detected communities [60], [61]. The communitude of a community, $\mathcal{C}_k$, is computed with respect to the supra-adjacency matrix as follows:

$$Comm(\mathcal{C}_k) = \frac{\frac{E_{intra}^{\mathcal{C}_k}}{E} - (\frac{E_{intra}^{\mathcal{C}_k} + E_{inter}^{\mathcal{C}_k}}{2E})^2}{(\frac{E_{intra}^{\mathcal{C}_k} + E_{inter}^{\mathcal{C}_k}}{2E})^2 (1 - (\frac{E_{intra}^{\mathcal{C}_k} + E_{inter}^{\mathcal{C}_k}}{2E})^2)}, \quad (25)$$

  where $E_{intra}^{\mathcal{C}_k}$ is the sum of the edges that connect all the nodes within the community $\mathcal{C}_k$, $E_{inter}^{\mathcal{C}_k}$ is the sum of edges that connect the nodes in community $\mathcal{C}_k$ to nodes from other communities and $2E$ is the total edge weight of the supra-adjacency matrix. The communitude metric has an upper bound of 1.
- In Step 11 of Algorithm 1, the detected communities are sorted in a descending order based on their communitude values to determine the best set of communities. In the proposed approach, each node can belong to either an intra- or inter-layer community. Consequently, the communitude values of the intra- and inter-layer communities that the node belongs to are compared and the community with the higher value is chosen.
- The best set of K communities for the given multi-layer network is determined as the set with the highest communitude values, $\mathcal{C}_{\mathcal{M}} = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_K\}$.

E. COMPUTATIONAL COMPLEXITY OF THE PROPOSED ALGORITHM

In the proposed approach, the variables $\mathbf{U}_\alpha$, $\mathbf{B}_\alpha$ where $\alpha \in \{1, 2\}$ and $\mathbf{S}^{12}$ are updated using the multiplicative update algorithm. The nonnegative embeddings, $\mathbf{U}_\alpha$ and $\mathbf{B}_\alpha$, are initialized using NNDSVD with complexity $\mathcal{O}(r_\alpha n^{\alpha^2})$ and $\mathcal{O}(r_{\alpha\beta} n^\alpha n^\beta)$, respectively. The AS metric is then used to determine the initial number of communities in each intra-layer network over a range of communities, $\{2, 3, \ldots, K_{max}\}$. The computational complexity for computing AS is $K_{max}\mathcal{O}(|E_\alpha|)$, $\alpha \in \{1, 2\}$ where $|E_\alpha|$ is the number of edges in the α-th layer. For each iteration, the updates of $\mathbf{U}_\alpha$ and $\mathbf{B}_\alpha$ have a complexity of $\mathcal{O}(r_\alpha n^{\alpha^2})$ and $\mathcal{O}(r_{\alpha\beta} n^\alpha n^\beta + r_\alpha n^{\alpha^2})$, respectively. Moreover, the update of $\mathbf{S}^{12}$ has a complexity of $\mathcal{O}(r_{\alpha\beta} n^\alpha n^\beta)$. Therefore, for a 2-layer network with the total number of iterations required until convergence equal to l_{max}, the total complexity of the proposed approach is approximately equal to $2l_{max}\mathcal{O}(r_\alpha \max(n^\alpha, n^\beta)^2)$.

[2]https://www.boutsidis.org/software.html

Algorithm 1 Community Detection in **M**ulti-**L**ayer Networks: **J**oint **N**onegative **M**atrix **F**actorization (ML-JNMF)

Input: $\mathbf{A}^1$, $\mathbf{A}^2$, $\mathbf{A}^{12}$, μ_1, μ_2
Output: $\mathbf{U}_1$, $\mathbf{U}_2$, $\mathbf{B}_1$, $\mathbf{B}_2$, $\mathbf{S}^{12}$, $\mathcal{C}_{intra}$, $\mathcal{C}_{inter}$
1: Initialize $\mathbf{U}_1$, $\mathbf{U}_2$, $\mathbf{Z}^1$, $\mathbf{Z}^2$, $\mathbf{B}_1$, $\mathbf{B}_2$, $\mathbf{S}^{12}$, $\mathbf{Z}^{12}$
2: **while** not converge **do**
3: Update $\mathbf{U}_1$ using Eq. 21.
4: Update $\mathbf{U}_2$ using Eq. 21.
5: Update $\mathbf{B}_1$ using Eq. 22.
6: Update $\mathbf{B}_2$ using Eq. 23.
7: Update $\mathbf{S}^{12}$ using Eq. 24.
8: **end while**
9: **for** $i = 1 : n^\alpha$ **do**
10: $j^* = \max_j \mathbf{U}_\alpha(i, j)$ %Intra-layer communities
11: **end for**
12: **for** $i = 1 : n^\alpha$ **do**
13: $j^* = \max_j \mathbf{B}_\alpha(i, j)$ %Inter-layer communities
14: **end for**
15: Determine inter-layer communities by factorizing $\mathbf{S}^{12}$ using NMF.
16: Calculate the quality of the communities using communitude metric defined in Eq. 25.
17: Determine the best communities.

F. CONVERGENCE ANALYSIS OF THE PROPOSED APPROACH

To prove the convergence of the update rule of $\mathbf{U}_\alpha$ in Eq. 21, the auxiliary function method is adopted [62]. Following [62], an auxiliary function can be defined as follows.

Definition 3: $\mathcal{L}(\mathbf{U}_\alpha, \mathbf{U}_\alpha^t)$ is an auxiliary function of $\mathcal{J}(\mathbf{U}_\alpha)$, if it satisfies the following conditions:

$$\mathcal{L}(\mathbf{U}_\alpha, \mathbf{U}_\alpha^t) \geq \mathcal{J}(\mathbf{U}_\alpha), \quad \mathcal{L}(\mathbf{U}_\alpha, \mathbf{U}_\alpha) = \mathcal{J}(\mathbf{U}_\alpha) \quad (26)$$

The concept of the auxiliary function is helpful because of the following lemma:

Lemma 1: If $\mathcal{L}$ is an auxiliary function, then $\mathcal{J}(\mathbf{U}_\alpha)$ is non-increasing under the update:

$$\mathbf{U}_\alpha{}^{t+1} = \underset{\mathbf{U}_\alpha}{\operatorname{argmin}} \mathcal{L}(\mathbf{U}_\alpha, \mathbf{U}_\alpha^t). \quad (27)$$

The proof of Lemma 1 is given in [62].

Theorem 1: Fixing $\mathbf{B}_\alpha$ *with* $\alpha \in \{1,2\}$ *and* $\mathbf{S}^{12}$*, the Lagrangian function* $\mathcal{J}(\mathbf{U}_\alpha)$ *is non-increasing under the update rule in Eq. 21.*

Proof: Let $\mathcal{J}(u_\alpha)$ be the part of $\mathcal{J}(\mathbf{U}_\alpha)$ that is dependent on $(u_\alpha)_{ij}$, using Eq. 36, we get:

$$\mathcal{J}'(u_\alpha) = \big(-2\mathbf{Z}^\alpha\mathbf{A}^\alpha\mathbf{U}_\alpha - 2\mathbf{A}^\alpha\mathbf{Z}^\alpha\mathbf{U}_\alpha + 2\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{Z}^\alpha\mathbf{U}_\alpha + 2\mathbf{Z}^\alpha\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{U}_\alpha + 4\mu_2\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{U}_\alpha - 4\mu_2\mathbf{B}_\alpha\mathbf{B}_\alpha^\top\mathbf{U}_\alpha\big)_{ij}. \quad (28)$$

The second-order derivative of $\mathcal{J}'(u_\alpha)$ with respect to $(u_\alpha)_{ij}$ is then equal to:

$$\begin{aligned}\mathcal{J}''(u_\alpha) = &-2\big(\mathbf{Z}^\alpha\mathbf{A}^\alpha + \mathbf{A}^\alpha\mathbf{Z}^\alpha\big)_{ii} + 2\Big[\big(\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{Z}^\alpha\big)_{ii} \\ &+ \big(\mathbf{U}_\alpha^\top\mathbf{Z}^\alpha\big)_{ij}\big(u_\alpha\big)_{ij} + z_{ii}^\alpha\sum_p\big(u_\alpha\big)_{pj}^2\Big] \\ &+2\Big[\big(\mathbf{Z}^\alpha\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\big)_{ii} + \big(\mathbf{Z}^\alpha\mathbf{U}_\alpha\big)_{ij}\big(u_\alpha\big)_{ij} \\ &+ z_{ii}^\alpha\sum_p\big(u_\alpha\big)_{pj}^2\Big] + 4\mu_2\Big[\big(\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\big)_{ij} \\ &+ \big(\mathbf{U}_\alpha\big)_{ij}\big(u_\alpha\big)_{ij} + \sum_p\big(u_\alpha\big)_{pj}^2\Big] - 4\mu_2\big(\mathbf{B}_\alpha\mathbf{B}_\alpha^\top\big)_{ij}. \quad (29)\end{aligned}$$

Let $(u_\alpha)_{ij}^t$ denote the t^{th} iterative update value of $(u_\alpha)_{ij}$, the Taylor series expansion of $\mathcal{J}(u_\alpha)$ at $(u_\alpha)_{ij}^t$ can be then formulated as:

$$\mathcal{J}(u_\alpha) = \mathcal{J}\big((u_\alpha)_{ij}^t\big) + \mathcal{J}'\big((u_\alpha)_{ij}^t\big)\big(u_\alpha - (u_\alpha)_{ij}^t\big) + \frac{1}{2}\mathcal{J}''\big((u_\alpha)_{ij}^t\big)\big(u_\alpha - (u_\alpha)_{ij}^t\big)^2. \quad (30)$$

We define the following $\mathcal{L}(u_\alpha, (u_\alpha)_{ij}^t)$ as an auxiliary function of $\mathcal{J}(u_\alpha)$:

$$\begin{aligned}&\mathcal{L}(u_\alpha, (u_\alpha)_{ij}^t) \\ &= \mathcal{J}\big((u_\alpha)_{ij}^t\big) + 3\mathcal{J}'\big((u_\alpha)_{ij}^t\big)\big(u_\alpha - (u_\alpha)_{ij}^t\big) \\ &\quad + \frac{1}{2}\Bigg[6\frac{\big(\mathbf{U}_\alpha^t\mathbf{U}_\alpha^{t\top}\mathbf{Z}^\alpha\mathbf{U}_\alpha^t + \mathbf{Z}^\alpha\mathbf{U}_\alpha^t\mathbf{U}_\alpha^{t\top}\mathbf{U}_\alpha^t\big)_{ij}}{(u_\alpha)_{ij}^t} \\ &\quad + 12\frac{\big(\mu_2\mathbf{U}_\alpha^t\mathbf{U}_\alpha^{t\top}\mathbf{U}_\alpha^t\big)_{ij}}{(u_\alpha)_{ij}^t}\Bigg]\big(u_\alpha - (u_\alpha)_{ij}^t\big)^2. \quad (31)\end{aligned}$$

The proof that the function $\mathcal{L}(u_\alpha, (u_\alpha)_{ij}^t)$ in Eq. 31 is an auxiliary function of $\mathcal{J}(u_\alpha)$ is shown in the Appendix.

In line with Lemma 1, we need to find the minimum of $\mathcal{L}(u_\alpha, (u_\alpha)_{ij}^t)$ with respect to u_α which can be determined by setting the gradient to zero:

$$\begin{aligned}&\frac{\partial\mathcal{L}(u_\alpha, (u_\alpha)_{ij}^t)}{\partial u_\alpha} \\ &= 3\mathcal{J}'\big((u_\alpha)_{ij}^t\big) + 6\frac{\big(\mathbf{U}_\alpha^t\mathbf{U}_\alpha^{t\top}\mathbf{Z}^\alpha\mathbf{U}_\alpha^t\big)_{ij}}{(u_\alpha)_{ij}^t}\big(u_\alpha - (u_\alpha)_{ij}^t\big) \\ &\quad + 6\frac{\big(\mathbf{Z}^\alpha\mathbf{U}_\alpha^t\mathbf{U}_\alpha^{t\top}\mathbf{U}_\alpha^t\big)_{ij}}{(u_\alpha)_{ij}^t}\big(u_\alpha - (u_\alpha)_{ij}^t\big) \\ &\quad + 12\frac{\big(\mu_2\mathbf{U}_\alpha^t\mathbf{U}_\alpha^{t\top}\mathbf{U}_\alpha^t\big)_{ij}}{(u_\alpha)_{ij}^t}\big(u_\alpha - (u_\alpha)_{ij}^t\big) = 0. \quad (32)\end{aligned}$$

Substituting $\mathcal{J}'((u_\alpha^t)_{ij})$ from Eq. 28 in Eq. 32 and simplifying the equation by cancelling the common terms, we get:

$$\begin{aligned}&3\big(-2\mathbf{Z}^\alpha\mathbf{A}^\alpha\mathbf{U}_\alpha^t - 2\mathbf{A}^\alpha\mathbf{Z}^\alpha\mathbf{U}_\alpha^t - 4\mu_2\mathbf{B}_\alpha\mathbf{B}_\alpha^\top\mathbf{U}_\alpha^t\big)_{ij} \\ &\quad + 3\Big[2\big(\mathbf{U}_\alpha^t\mathbf{U}_\alpha^{t\top}\mathbf{Z}^\alpha\mathbf{U}_\alpha^t\big)_{ij} + 2\big(\mathbf{Z}^\alpha\mathbf{U}_\alpha^t\mathbf{U}_\alpha^{t\top}\mathbf{U}_\alpha^t\big)_{ij} \\ &\quad + 4\big(\mu_2\mathbf{U}_\alpha^t\mathbf{U}_\alpha^{t\top}\mathbf{U}_\alpha^t\big)_{ij}\Big]\frac{u_\alpha}{(u_\alpha^t)_{ij}} = 0. \quad (33)\end{aligned}$$

Finally, replacing u_α by $(u_\alpha^{t+1})_{ij}$, we obtain the same update rule in Eq. 21 as:

$$(u_\alpha^{t+1})_{ij} = \frac{\big(\mathbf{Z}^\alpha\mathbf{A}^\alpha\mathbf{U}_\alpha^t + \mathbf{A}^\alpha\mathbf{Z}^\alpha\mathbf{U}_\alpha^t + \mu_2'\mathbf{B}_\alpha\mathbf{B}_\alpha^\top\mathbf{U}_\alpha^t\big)_{ij}(u_\alpha^t)_{ij}}{\big(\mathbf{U}_\alpha^t\mathbf{U}_\alpha^{t\top}\mathbf{Z}^\alpha\mathbf{U}_\alpha^t + \mathbf{Z}^\alpha\mathbf{U}_\alpha^t\mathbf{U}_\alpha^{t\top}\mathbf{U}_\alpha^t + \mu_2'\mathbf{U}_\alpha^t\mathbf{U}_\alpha^{t\top}\mathbf{U}_\alpha^t\big)_{ij}}. \quad (34)$$

□

Similar procedure can be followed to prove the convergence of the other update rules. Consequently, their proofs of convergence will not be presented explicitly in this paper.

V. RESULTS

The performance of the proposed algorithm is evaluated on multiple simulated and real-world multi-layer networks. All experiments are performed using MATLAB R2020b on a desktop with the specifications (Intel(R) Core(TM) i7-9700 CPU @ 3.00GHz 3.00 GHz and RAM of 16GB). The quality of the detected community structure is evaluated through multiple evaluation metrics with respect to the ground truth including normalized mutual information (NMI), variation of information (VI), F-value, recall and purity. All evaluation metrics are normalized between [0, 1]. High values of NMI, F-value, recall and purity and low values of VI indicate better community detection results.

The performance of the proposed approach is compared to existing methods including symmetric nonnegative matrix factorization[3] (SymNMF) [44], [45], orthogonal nonnegative matrix tri-factorization[4] (ONMTF) [57], block spectral clustering[5] (BLSC) [30], generalized Louvain[6] (GenLov) [53], collective symmetric nonnegative matrix factorization (CSNMF) [37], collective projective nonnegative matrix factorization (CPNMF) [37], collective symmetric nonnegative matrix tri-factorization (CSNMTF) [37], geodesic density gradient[7] (GDG) algorithm [63], subspace based community detection with fusion[8] (SSCF) [64] and multiplex cellular communities for tissue phenotyping (MCTP) [38]. SymNMF, ONMTF, BLSC, GenLov, GDG

[3] https://github.com/hiroyuki-kasai/NMFLibrary
[4] https://sites.google.com/site/nmftool/home
[5] https://hkumath.hku.hk/ mng/myresearch
[6] https://sites.google.com/site/bctnet/
[7] https://www.researchgate.net/publication/323336660_Matlab_Code_of_Using_Geodesic_Space_Density_Gradients_for_Network_Community_Detection
[8] https://www.researchgate.net/publication/308377705_subSpaceComDet

and SSCF are applied to the supra-adjacency of the multi-layer network, whereas CSNMF, CPNMF, CSNMTF and MCTP are applied to the multiplex version of the multi-layer network. The performance of the different existing algorithms is compared to the proposed algorithm using the evaluation metrics mentioned above. The number of communities for SymNMF and ONMTF is determined based on dispersion coefficient[9] [51], [52], for GDG, SSCF, CSNMF, CPNMF, CSNMTF and MCTP by modularity whereas the eigengap criterion is used for BLSC.

A. SIMULATED MULTI-LAYER NETWORKS

1) SIMULATED WEIGHTED MULTI-LAYER NETWORKS

A set of undirected weighted two-layer networks are generated to evaluate the performance of the proposed approach. The edges of the networks are randomly selected from a truncated Gaussian distribution in the range of [0, 1]. The parameters of the generated networks are shown in Table 2. The networks are constructed with different community structures and number of communities. Moreover, different levels of sparse noise ($sn\%$) is randomly added to the network to evaluate the robustness of the different algorithms against outliers. In addition, a percent of inter-community edges are set randomly to zero ($ze\%$) to control the amount of mixing between communities. More specifically, when $ze\% = 0$, the network is fully connected and as $ze\%$ increases the communities become more distinct. In the implementation of the proposed community detection algorithm for weighted networks, a node's community membership is determined by k-means instead of using the largest entry in each row of the basis matrices. This is done since the entries of the basis matrices tend to be very close to each other and the maximum may not always give the correct community assignment.

The performance of the proposed approach in detecting the community structure of weighted multi-layer networks (WMLNs) is compared to other algorithms as presented in Table 3. The experiments are repeated 50 times and the comparison is conducted in terms of average NMI, VI, F-value, recall and purity. The regularization parameters are selected by searching the best values in the (μ_1, μ_2) grid. As it can be seen in Table 3, the proposed approach outperforms the other algorithms in terms of the different evaluation metrics. The performance of BLSC, GenLov, SymNMF and ONMTF is relatively good for WMLN 1 since it consists of two large inter-layer communities. However, the performance of these algorithms decays as the size of the inter-layer communities decreases as in WMLNs $2-6$ and as the sparse noise increases as in WMLN $4-6$. Moreover, GDG and SSCF algorithms show poor performance over all the networks since they cannot handle weighted networks. On the contrary, the proposed algorithm is robust to noise and outliers. As it can be seen in Table 3, the proposed algorithm is capable of detecting the community structure in the different networks accurately, such as fully connected networks with $ze\% = 0$ as WMLN 2 and 6, or noisy networks as in WMLN 4 and 6. In addition, the proposed algorithm performs better than the other algorithms in detecting the community structure of WMLN 5 where the community structure is composed of a large number of small communities. Finally, the proposed approach can handle multi-layer networks with variable number of nodes across layers as in WMLN 3 and 6, unlike CSNMF, CPNMF, CSNMTF and MTCP.

2) SIMULATED GIRVAN-NEWMAN MULTI-LAYER NETWORKS

In order to evaluate the performance of the proposed algorithm in detecting the community structure in binary multi-layer networks, a Girvan-Newman (GN) multi-layer network (GNMLN) is constructed by extending the model for single-layer networks [65]. Each layer in the the constructed network consists of 128 nodes and each node has an average degree of 16. Nodes in each layer are partitioned into 4 communities with 32 nodes each. The community structure is varied using the parameter z_{out}, which determines the number of external edges for each node, i.e., the number of edges that connects a node to nodes from other communities. In particular, as z_{out} decreases, the communities become more distinct. Moreover, the probability of a community to be an inter-layer community is determined by a user defined parameter, $a \in [0, 1]$. Particularly, when $a = 0$ the multi-layer network consists of only intra-layer communities whereas when $a = 1$ the multi-layer network consists of only inter-layer communities. In particular, the probability of a community to be an inter-layer community is set to $1 - \sqrt[M-1]{(1-a)}$ and the probability of intra- and inter-community edges are defined as $p_{in} = (16 - z_{out})/32$ and $p_{out} = z_{out}/96$, respectively.

The proposed approach is applied to a set of Girvan-Newman multi-layer networks with $M = 2$ and the parameters given in Table 4. The simulations are repeated 50 times. As it can be seen in the table, the proposed algorithm performs better than the other algorithms in detecting the community structure of the underlying network in terms of the different quality metrics. For instance, in GNMLNs $1-3$, where $z_{out} = 5$, the communities are more distinct compared to GNMLN 4 and 5. Consequently, the existing algorithms perform relatively better in detecting the community structure of the former. In addition, we notice that the performance of the existing algorithms is affected by the parameter a, with the performance decaying as a decreases. On the other hand, the proposed algorithm performs well for a variety of network structures. In particular, the proposed algorithm can detect the community structure even when z_{out} is increased. Moreover, it is not sensitive to the parameter a and can detect both intra- and inter-layer communities. Finally, it can be seen that GenLov algorithm performs slightly better than the proposed algorithm in detecting the community structure of GNMLN 5. This is because GenLov algorithm relies on modularity maximization which is known to perform better when the network consists of large communities. However, Genlov fails to detect the network's community structure

[9] https://sites.google.com/site/nmftool/home

TABLE 2. Parameters and ground truth community structure of the simulated weighted multi-layer networks (WMLNs) including: 1) Percent of zero inter-community edges (*ze*%); 2) Percent of randomly added sparse noise (*sn*%); 3) The number of nodes in each layer (n^1, n^2); 4) number of communities (NOC); 5) Ground truth communities; 6) Mean and standard deviation of intra- and inter-community edges for both intra- and inter-layer graphs.

Network	n^1, n^2	NOC	Ground truth communities	$\mu_{intra}, \sigma_{intra}, \mu_{inter}, \sigma_{inter}$
WMLN 1 $ze\% = 5$ $sn\% = 5$	100, 100	2	**Layer**1: $\mathcal{C}_1^1(1-40), \mathcal{C}_2^1(41-100)$ **Layer**2: $\mathcal{C}_1^2(1-70), \mathcal{C}_2^2(71-100)$ **community structure**: $\mathcal{C}_1^{12} = \{C_1^1, C_1^2\}, \mathcal{C}_2^{12} = \{C_2^1, C_2^2\}$	L^1: (0.8, 0.4, 0.3, 0.2) L^2: (0.5, 0.1, 0.2, 0.2) L^{12}: (0.4, 0.1, 0.3, 0.1)
WMLN 2 $ze\% = 0$ $sn\% = 5$	100, 100	7	**Layer**1: $\mathcal{C}_1^1(1-20), \mathcal{C}_2^1(21-40), \mathcal{C}_3^1(41-60), \mathcal{C}_4^1(61-100)$ **Layer**2: $\mathcal{C}_1^2(1-20), \mathcal{C}_2^2(21-35), \mathcal{C}_3^2(36-50), \mathcal{C}_4^2(51-75), \mathcal{C}_5^2(76-100)$ **community structure**: $\mathcal{C}_1^{12} = \{C_1^1, C_1^2\}, \mathcal{C}_2^{12} = \{C_2^1, C_2^2\}, \mathcal{C}_3^1, \mathcal{C}_4^1, \mathcal{C}_3^2, \mathcal{C}_4^2, \mathcal{C}_5^2$	L^1: (0.5, 0.3, 0.3, 0.2) L^2: (0.5, 0.2, 0.2, 0.2) L^{12}: (0.8, 0.2, 0.3, 0.1)
WMLN 3 $ze\% = 10$ $sn\% = 5$	100, 64	3	**Layer**1: $\mathcal{C}_1^1(1-50), \mathcal{C}_2^1(51-100)$ **Layer**2: $\mathcal{C}_1^2(1-32), \mathcal{C}_2^2(33-64)$ **community structure**: $\mathcal{C}_1^{12} = \{C_1^1, C_1^2\}, \mathcal{C}_2^1, \mathcal{C}_2^2$	L^1: (0.5, 0.4, 0.3, 0.2) L^2: (0.4, 0.1, 0.2, 0.2) L^{12}: (0.6, 0.2, 0.3, 0.2)
WMLN 4 $ze\% = 10$ $sn\% = 15$	100, 100	4	**Layer**1: $\mathcal{C}_1^1(1-40), \mathcal{C}_2^1(41-100)$ **Layer**2: $\mathcal{C}_1^2(1-30), \mathcal{C}_2^2(31-60), \mathcal{C}_3^2(61-100)$ **community structure**: $\mathcal{C}_1^{12} = \{C_1^1, C_1^2\}, \mathcal{C}_2^1, \mathcal{C}_2^2, \mathcal{C}_3^2$	L^1: (0.5, 0.4, 0.3, 0.2) L^2: (0.5, 0.1, 0.2, 0.2) L^{12}: (0.5, 0.2, 0.3, 0.1)
WMLN 5 $ze\% = 10$ $sn\% = 10$	100, 100	8	**Layer**1: $\mathcal{C}_1^1(1-20), \mathcal{C}_2^1(21-40), \mathcal{C}_3^1(41-60), \mathcal{C}_4^1(61-100)$ **Layer**2: $\mathcal{C}_1^2(1-15), \mathcal{C}_2^2(16-35), \mathcal{C}_3^2(36-50), \mathcal{C}_4^2(51-75), \mathcal{C}_5^2(76-100)$ **community structure**: $\mathcal{C}_1^{12} = \{C_1^1, C_1^2\}, \mathcal{C}_2^1, \mathcal{C}_3^1, \mathcal{C}_4^1, \mathcal{C}_2^2, \mathcal{C}_3^2, \mathcal{C}_4^2, \mathcal{C}_5^2$	L^1: (0.5, 0.4, 0.3, 0.2) L^2: (0.5, 0.1, 0.2, 0.2) L^{12}: (0.5, 0.2, 0.3, 0.1)
WMLN 6 $ze\% = 0$ $sn\% = 15$	100, 64	3	**Layer**1: $\mathcal{C}_1^1(1-50), \mathcal{C}_2^1(51-100)$ **Layer**2: $\mathcal{C}_1^2(1-32), \mathcal{C}_2^2(33-64)$ **community structure**: $\mathcal{C}_1^{12} = \{C_1^1, C_1^2\}, \mathcal{C}_2^1, \mathcal{C}_2^2$	L^1: (0.5, 0.4, 0.3, 0.2) L^2: (0.4, 0.1, 0.2, 0.2) L^{12}: (0.6, 0.2, 0.3, 0.2)

when a decreases which leads to smaller communities, as in GNMLN 4.

B. REGULARIZATION PARAMETER SELECTION

In the proposed approach, two regularization parameters are used, μ_1 and μ_2. In particular, μ_1 controls the inter-layer low-rank approximation term while μ_2 penalizes the similarity between intra- and inter-layer nonnegative embeddings. The effect of the regularization parameters on the performance of the proposed algorithm is studied through experimental validation. In particular, we found that the proposed algorithm is insensitive to the different regularization parameters values when the network is not noisy and consists of distinct communities. Moreover, we noticed that the algorithm is less sensitive to μ_1 than μ_2 and performs well under different values of μ_1 in noisy networks. On the other hand, μ_2 can be selected depending on the multi-layer network under study. More specifically, if the network tends to consist of intra-layer communities only, small values of μ_2 are recommended whereas if the network consists of only inter-layer communities or a mixture of intra- and inter-layer communities, larger values of μ_2 are recommended.

C. SCALABILITY ANALYSIS

In order to test the scalability of the proposed approach, a set of weighted multi-layer networks with different sizes are constructed. The size of the networks is varied from 32 to 8192 on a logarithmic scale. The networks are constructed with the intra- and inter-community edges randomly selected from a truncated Gaussian distribution in the range of [0, 1] with: $\mu^1_{intra} = 0.5$, $\sigma^1_{intra} = 0.1$, $\mu^1_{inter} = 0.3$, $\sigma^1_{inter} = 0.2$, $\mu^2_{intra} = 0.7$, $\sigma^2_{intra} = 0.1$, $\mu^2_{inter} = 0.2$, $\sigma^2_{inter} = 0.2$, $\mu^{12}_{intra} = 0.58$, $\sigma^{12}_{intra} = 0.1$, $\mu^{12}_{inter} = 0.3$, $\sigma^{12}_{inter} = 0.2$. The network consists of two layers with n^1 and n^2 and each layer consists of two equal size communities: $\mathcal{C}_1^1$ and $\mathcal{C}_2^1$ in layer 1 and $\mathcal{C}_1^2$ and $\mathcal{C}_2^2$ in layer 2. The community structure of the multi-layer network consists of $\mathcal{C}_1^{12} = \{\mathcal{C}_1^1, \mathcal{C}_1^2\}$, $\mathcal{C}_2^1$ and $\mathcal{C}_2^2$. The number of communities is given as an input for each one of the algorithms. The run time of the different algorithms with respect to the variation of the network's size is shown in Fig. 2. In addition, a performance comparison between the different methods in detecting the community structure is presented in Table 5 in terms of the different quality metrics. As it can be seen from Fig. 2, the different methods are log-linear as the number of nodes increases. In addition, the proposed algorithm is faster than SymNMF, GDG and SSCF with increasing number of nodes. On the other hand, the remaining methods are faster compared to the proposed approach. However, as it can be seen from Table 5, the proposed algorithm maintains its good performance in detecting the community structure as the network's size increases.

D. REAL-WORLD MULTI-LAYER NETWORKS

To evaluate the performance of the proposed algorithm in detecting the community structure of real multi-layer networks, multiple networks from different disciplines are adopted. In this section, a brief description of the tested real-world networks is provided.

TABLE 3. Performance comparison between the proposed method (ML-JNMF), block spectral clustering (BLSC), generalized louvain (GenLov), symmetric nonnegative matrix factorization (SymNMF), orthogonal nonnegative matrix tri-factorization (ONMTF), collective symmetric nonnegative matrix factorization (CSNMF), collective projective nonnegative matrix factorization (CSNMF), collective symmetric nonnegative matrix tri-factorization (CSNMTF), Geodesic Density Gradient (GDG) algorithm, subspace based community detection with fusion (SSCF) and multiplex cellular communities for tissue phenotyping (MCTP) in detecting the community structure in weighted multi-layer networks (WMLNs) given in Table 2. The comparison is conducted in terms of average NMI, VI, F-value, recall and purity.

Network	Method	NMI	VI	F-value	Recall	Purity
WMLN 1 $n = 200$	BLSC	0.9066 ± 0.1750	0.0202 ± 0.0245	0.9066 ± 0.1750	0.9073 ± 0.1762	0.9770 ± 0.0800
	GenLov	0.8684 ± 0.2907	0.0550 ± 0.1387	0.8674 ± 0.2936	0.8692 ± 0.2939	0.9542 ± 0.1248
	SymNMF	0.7428 ± 0.2578	0.1017 ± 0.1419	0.7235 ± 0.2981	0.7241 ± 0.2986	0.9525 ± 0.0547
	ONMTF	0.8783 ± 0.2051	0.0390 ± 0.0683	0.8789 ± 0.2042	0.9065 ± 0.1681	0.9770 ± 0.0558
	CSNMF	0.3808 ± 0.0528	0.1632 ± 0.0391	0.3820 ± 0.0481	0.3768 ± 0.0355	0.8377 ± 0.0168
	CPNMF	0.2389 ± 0.0535	0.2812 ± 0.0351	0.2278 ± 0.0943	0.2812 ± 0.1171	0.8060 ± 0.0392
	CSNMTF	0.3695 ± 0.0631	0.1662 ± 0.0358	0.3704 ± 0.0613	0.3665 ± 0.0435	0.8287 ± 0.0159
	GDG	0.0227 ± 0.0107	0.2759 ± 0.0466	0.0219 ± 0.0112	0.0235 ± 0.0139	0.5953 ± 0.0217
	SSCF	0.0092 ± 0.0099	0.3166 ± 0.0781	0.0084 ± 0.0110	0.0090 ± 0.0125	0.5638 ± 0.0222
	MTCP	0.4002 ± 0.0937	0.1725 ± 0.0593	0.3852 ± 0.1317	0.3857 ± 0.1205	0.8337 ± 0.0203
	ML-JNMF	$\mathbf{0.9609 \pm 0.0351}$	$\mathbf{0.0102 \pm 0.0091}$	$\mathbf{0.9609 \pm 0.0351}$	$\mathbf{0.9609 \pm 0.0354}$	$\mathbf{0.9947 \pm 0.0054}$
WMLN 2 $n = 200$	BLSC	0.4451 ± 0.0989	0.2843 ± 0.1033	0.4140 ± 0.1363	0.2737 ± 0.1096	0.4628 ± 0.1605
	GenLov	0.7835 ± 0.0452	0.1328 ± 0.0242	0.7835 ± 0.0452	0.6787 ± 0.0647	0.7293 ± 0.0547
	SymNMF	0.4257 ± 0.1627	0.3085 ± 0.0767	0.4205 ± 0.1598	0.3269 ± 0.1461	0.4870 ± 0.1141
	ONMTF	0.4645 ± 0.1290	0.3018 ± 0.0803	0.4597 ± 0.1308	0.3660 ± 0.1231	0.5025 ± 0.0898
	CSNMF	0.6189 ± 0.0417	0.2082 ± 0.0196	0.6189 ± 0.0417	0.4752 ± 0.0491	0.5800 ± 0.0216
	CPNMF	0.6120 ± 0.0329	0.2117 ± 0.0161	0.6120 ± 0.0329	0.4685 ± 0.0378	0.5772 ± 0.0125
	CSNMTF	0.6070 ± 0.0436	0.2184 ± 0.0229	0.6070 ± 0.0436	0.4742 ± 0.0543	0.5788 ± 0.0213
	GDG	0.1955 ± 0.0340	0.4053 ± 0.0243	0.1955 ± 0.0340	0.1386 ± 0.0292	0.2870 ± 0.0179
	SSCF	0.0279 ± 0.0192	0.5149 ± 0.0664	0.0279 ± 0.0192	0.0223 ± 0.0185	0.2373 ± 0.0228
	MTCP	0.6406 ± 0.0348	0.2011 ± 0.0137	0.6406 ± 0.0348	0.5050 ± 0.0568	0.5892 ± 0.0257
	ML-JNMF	$\mathbf{0.8682 \pm 0.0132}$	$\mathbf{0.0904 \pm 0.0088}$	$\mathbf{0.8682 \pm 0.0133}$	$\mathbf{0.8334 \pm 0.0275}$	$\mathbf{0.8398 \pm 0.0317}$
WMLN 3 $n = 164$	BLSC	0.5669 ± 0.0376	0.2905 ± 0.0846	0.4347 ± 0.3090	0.3483 ± 0.2562	0.9482 ± 0.1151
	GenLov	0.8046 ± 0.0058	0.0659 ± 0.0019	0.8046 ± 0.0058	0.6737 ± 0.0048	0.8049 ± 0.0000
	SymNMF	0.7923 ± 0.0364	0.0701 ± 0.0122	0.7923 ± 0.0364	0.6633 ± 0.0307	0.8033 ± 0.0089
	ONMTF	0.8046 ± 0.0058	0.0659 ± 0.0019	0.8046 ± 0.0058	0.6737 ± 0.0048	0.8049 ± 0.0000
	GDG	0.0053 ± 0.0042	0.3426 ± 0.0418	0.0052 ± 0.0041	0.0045 ± 0.0036	0.5000 ± 0.0000
	SSCF	0.0216 ± 0.0157	0.4300 ± 0.1083	0.0162 ± 0.0117	0.0161 ± 0.0124	0.5106 ± 0.0158
	ML-JNMF	$\mathbf{0.8777 \pm 0.0419}$	$\mathbf{0.0569 \pm 0.0203}$	$\mathbf{0.8777 \pm 0.0419}$	$\mathbf{0.9937 \pm 0.0195}$	$\mathbf{0.9986 \pm 0.0044}$
WMLN 4 $n = 200$	BLSC	0.3736 ± 0.1154	0.4897 ± 0.2627	0.1902 ± 0.1931	0.1411 ± 0.1537	0.7688 ± 0.2419
	GenLov	0.4421 ± 0.3121	0.3117 ± 0.1908	0.4345 ± 0.3207	0.4373 ± 0.3184	0.6987 ± 0.1929
	SymNMF	0.2484 ± 0.0609	0.4999 ± 0.0537	0.1542 ± 0.0702	0.1463 ± 0.0709	0.6140 ± 0.0528
	ONMTF	0.1954 ± 0.0995	0.5301 ± 0.1144	0.1551 ± 0.1162	0.1394 ± 0.0845	0.5475 ± 0.0433
	CSNMF	0.5219 ± 0.0610	0.1988 ± 0.0168	0.5219 ± 0.0610	0.4381 ± 0.0864	0.6288 ± 0.0137
	CPNMF	0.3729 ± 0.0610	0.2850 ± 0.0339	0.3729 ± 0.0610	0.3372 ± 0.0642	0.5862 ± 0.0285
	CSNMTF	0.4000 ± 0.0959	0.2600 ± 0.0623	0.4000 ± 0.0959	0.3406 ± 0.0902	0.5857 ± 0.0403
	GDG	0.2436 ± 0.0450	0.3025 ± 0.0319	0.2436 ± 0.0450	0.1928 ± 0.0365	0.4530 ± 0.0165
	SSCF	0.0211 ± 0.0135	0.4661 ± 0.0864	0.0180 ± 0.0108	0.0169 ± 0.0110	0.3730 ± 0.0174
	MTCP	0.3953 ± 0.0587	0.2423 ± 0.0424	0.3953 ± 0.0587	0.3106 ± 0.0391	0.5940 ± 0.0253
	ML-JNMF	$\mathbf{0.7550 \pm 0.0952}$	$\mathbf{0.1260 \pm 0.0444}$	$\mathbf{0.7534 \pm 0.0971}$	$\mathbf{0.7723 \pm 0.1193}$	$\mathbf{0.8650 \pm 0.1123}$
WMLN 5 $n = 200$	BLSC	0.3927 ± 0.2351	0.5390 ± 0.1113	0.0708 ± 0.0398	0.0401 ± 0.0237	0.7448 ± 0.3631
	GenLov	0.2030 ± 0.0791	0.5780 ± 0.0696	0.2030 ± 0.0791	0.1903 ± 0.0703	0.3722 ± 0.0707
	SymNMF	0.2790 ± 0.0506	0.5775 ± 0.0430	0.2602 ± 0.0525	0.2639 ± 0.0528	0.4355 ± 0.0538
	ONMTF	0.1351 ± 0.0301	0.7030 ± 0.0244	0.1229 ± 0.0273	0.1262 ± 0.0282	0.3205 ± 0.0321
	CSNMF	0.4622 ± 0.0811	0.3400 ± 0.0427	0.4622 ± 0.0811	0.3861 ± 0.0856	0.4460 ± 0.0564
	CPNMF	0.2415 ± 0.0825	0.4500 ± 0.0490	0.2415 ± 0.0825	0.1893 ± 0.0700	0.3245 ± 0.0402
	CSNMTF	0.2225 ± 0.0756	0.4620 ± 0.0603	0.2225 ± 0.0756	0.1775 ± 0.0776	0.3143 ± 0.0549
	GDG	0.2549 ± 0.0430	0.3963 ± 0.0277	0.2549 ± 0.0430	0.1779 ± 0.0362	0.2888 ± 0.0212
	SSCF	0.0354 ± 0.0202	0.5774 ± 0.0731	0.0352 ± 0.0198	0.0294 ± 0.0204	0.2242 ± 0.0165
	MTCP	0.2151 ± 0.0777	0.4294 ± 0.0646	0.2151 ± 0.0777	0.1539 ± 0.0548	0.3137 ± 0.0370
	ML-JNMF	$\mathbf{0.5739 \pm 0.0577}$	$\mathbf{0.2842 \pm 0.0378}$	$\mathbf{0.5739 \pm 0.0577}$	$\mathbf{0.5012 \pm 0.0664}$	$\mathbf{0.5443 \pm 0.0645}$
WMLN 6 $n = 164$	BLSC	0.4234 ± 0.1074	0.3936 ± 0.0890	0.1877 ± 0.2245	0.1511 ± 0.1914	0.8476 ± 0.1501
	GenLov	0.3227 ± 0.1781	0.3129 ± 0.1189	0.3174 ± 0.1814	0.3231 ± 0.1677	0.6941 ± 0.0744
	SymNMF	0.1757 ± 0.1166	0.5013 ± 0.1278	0.1261 ± 0.1352	0.1122 ± 0.1125	0.6437 ± 0.0448
	ONMTF	0.6314 ± 0.2138	0.1671 ± 0.1659	0.6182 ± 0.2436	0.5227 ± 0.1985	0.7705 ± 0.0583
	GDG	0.0068 ± 0.0068	0.3381 ± 0.0320	0.0068 ± 0.0068	0.0060 ± 0.0064	0.5002 ± 0.0011
	SSCF	0.0138 ± 0.0106	0.3929 ± 0.0806	0.0120 ± 0.0096	0.0120 ± 0.0109	0.5022 ± 0.0041
	ML-JNMF	$\mathbf{0.7318 \pm 0.0828}$	$\mathbf{0.1197 \pm 0.0368}$	$\mathbf{0.7273 \pm 0.0879}$	$\mathbf{0.7915 \pm 0.1216}$	$\mathbf{0.9333 \pm 0.0576}$

1) MOBILE PHONE NETWORKS

Two mobile phone networks, MIT Reality Mining,[10] [66] and Nokia Research Center (NRC) Lausanne [67], are adopted to test the performance of the proposed approach. Interactions in both networks represent three types of mobile phone communications between 87 users for the MIT network and 136 users for the NRC network. The interactions are the physical location, bluetooth scans and phone calls. The multi-layer network is constructed where the intra-layer graphs encode physical location and bluetooth scans and the

[10] http://reality.media.mit.edu/download.php

TABLE 4. Performance comparison between the proposed method (ML-JNMF), block spectral clustering (BLSC), generalized louvain (GenLov), symmetric nonnegative matrix factorization (SymNMF), orthogonal nonnegative matrix tri-factorization (ONMTF), collective symmetric nonnegative matrix factorization (CSNMF), collective projective nonnegative matrix factorization (CSNMF), collective symmetric nonnegative matrix tri-factorization (CSNMTF), Geodesic Density Gradient (GDG) algorithm, subspace based community detection with fusion (SSCF) and multiplex cellular communities for tissue phenotyping (MCTP) in detecting the community structure in binary Girvan-Newman multi-layer networks (GNMLNs). The comparison is conducted in terms of average NMI, VI, F-value, recall and purity.

Network	Method	NMI	VI	F-value	Recall	Purity
GNMLN 1	BLSC	0.5518 ± 0.1545	0.5298 ± 0.1859	0.1788 ± 0.2712	0.1438 ± 0.2591	0.8902 ± 0.2391
$n = 256$	GenLov	0.9786 ± 0.0209	0.0158 ± 0.0152	0.9786 ± 0.0209	0.9733 ± 0.0321	0.9763 ± 0.0429
$z_{out} = 5$	SymNMF	0.6315 ± 0.0736	0.2732 ± 0.0547	0.6315 ± 0.0736	0.6241 ± 0.0721	0.6023 ± 0.0923
$a = 0.0$	ONMTF	0.6113 ± 0.0702	0.2989 ± 0.0549	0.5956 ± 0.0765	0.5995 ± 0.0785	0.6404 ± 0.0817
	CSNMF	0.8000 ± 0.0000	0.1250 ± 0.0000	0.8000 ± 0.0000	0.6667 ± 0.0000	0.5000 ± 0.0000
	CPNMF	0.8000 ± 0.0000	0.1250 ± 0.0000	0.8000 ± 0.0000	0.6667 ± 0.0000	0.5000 ± 0.0000
	CSNMTF	0.7886 ± 0.0283	0.1319 ± 0.0154	0.7886 ± 0.0283	0.6574 ± 0.0321	0.4949 ± 0.0228
	GDG	0.9312 ± 0.0372	0.0509 ± 0.0264	0.9265 ± 0.0381	0.9174 ± 0.0553	0.9288 ± 0.0713
	SSCF	0.0365 ± 0.0215	0.6119 ± 0.0841	0.0351 ± 0.0194	0.0318 ± 0.0208	0.1887 ± 0.0275
	MTCP	0.7827 ± 0.0408	0.1344 ± 0.0200	0.7827 ± 0.0408	0.6498 ± 0.0508	0.4874 ± 0.0381
	ML-JNMF	$\mathbf{0.9957 \pm 0.0068}$	$\mathbf{0.0033 \pm 0.0052}$	$\mathbf{0.9957 \pm 0.0068}$	$\mathbf{0.9964 \pm 0.0052}$	$\mathbf{0.9982 \pm 0.0029}$
GNMLN 2	BLSC	0.5395 ± 0.1424	0.2702 ± 0.1725	0.4582 ± 0.2314	0.3298 ± 0.2158	0.5944 ± 0.2355
$n = 256$	GenLov	0.9931 ± 0.0096	0.0047 ± 0.0066	0.9931 ± 0.0096	0.9929 ± 0.0098	0.9971 ± 0.0041
$z_{out} = 5$	SymNMF	0.5556 ± 0.1609	0.2278 ± 0.0729	0.5556 ± 0.1609	0.4638 ± 0.1909	0.5339 ± 0.1458
$a = 0.5$	ONMTF	0.5304 ± 0.1713	0.2349 ± 0.0739	0.5276 ± 0.1699	0.4340 ± 0.2028	0.5503 ± 0.1626
	CSNMF	0.8582 ± 0.0610	0.0837 ± 0.0380	0.8582 ± 0.0610	0.7672 ± 0.0863	0.6914 ± 0.1301
	CPNMF	0.8707 ± 0.0500	0.0760 ± 0.0325	0.8707 ± 0.0500	0.7745 ± 0.0806	0.6958 ± 0.1300
	CSNMTF	0.8443 ± 0.0554	0.0912 ± 0.0344	0.8443 ± 0.0554	0.7554 ± 0.0873	0.6850 ± 0.1289
	GDG	0.9227 ± 0.0497	0.0494 ± 0.0323	0.9204 ± 0.0499	0.8944 ± 0.0757	0.9092 ± 0.0841
	SSCF	0.0287 ± 0.0187	0.5491 ± 0.0837	0.0273 ± 0.0165	0.0246 ± 0.0170	0.2684 ± 0.0405
	MTCP	0.8579 ± 0.0539	0.0847 ± 0.0356	0.8579 ± 0.0539	0.7721 ± 0.0808	0.6948 ± 0.1300
	ML-JNMF	$\mathbf{0.9996 \pm 0.0014}$	$\mathbf{0.0003 \pm 0.0009}$	$\mathbf{0.9996 \pm 0.0014}$	$\mathbf{1.0000 \pm 0.0000}$	$\mathbf{1.0000 \pm 0.0000}$
GNMLN 3	BLSC	$\mathbf{1.0000 \pm 0.0000}$	$\mathbf{0.0000 \pm 0.0000}$	$\mathbf{1.0000 \pm 0.0000}$	$\mathbf{1.0000 \pm 0.0000}$	$\mathbf{1.0000 \pm 0.0000}$
$n = 256$	GenLov	$\mathbf{1.0000 \pm 0.0000}$	$\mathbf{0.0000 \pm 0.0000}$	$\mathbf{1.0000 \pm 0.0000}$	$\mathbf{1.0000 \pm 0.0000}$	$\mathbf{1.0000 \pm 0.0000}$
$z_{out} = 5$	SymNMF	0.7011 ± 0.1927	0.1417 ± 0.0942	0.7011 ± 0.1927	0.6690 ± 0.1973	0.7650 ± 0.1504
$a = 1.0$	ONMTF	0.5802 ± 0.1624	0.2163 ± 0.0875	0.5559 ± 0.1755	0.5496 ± 0.1844	0.7090 ± 0.1398
	CSNMF	0.9961 ± 0.0148	0.0021 ± 0.0079	0.9961 ± 0.0148	1.0000 ± 0.0000	1.0000 ± 0.0000
	CPNMF	$\mathbf{1.0000 \pm 0.0000}$	$\mathbf{0.0000 \pm 0.0000}$	$\mathbf{1.0000 \pm 0.0000}$	$\mathbf{1.0000 \pm 0.0000}$	$\mathbf{1.0000 \pm 0.0000}$
	CSNMTF	0.9861 ± 0.0342	0.0070 ± 0.0165	0.9861 ± 0.0342	0.9901 ± 0.0458	0.9909 ± 0.0456
	GDG	0.9842 ± 0.0273	0.0084 ± 0.0146	0.9814 ± 0.0373	0.9944 ± 0.0285	0.9999 ± 0.0007
	SSCF	0.0211 ± 0.0112	0.4618 ± 0.0906	0.0180 ± 0.0090	0.0167 ± 0.0095	0.3178 ± 0.0267
	MTCP	0.9823 ± 0.0325	0.0087 ± 0.0154	0.9823 ± 0.0325	0.9839 ± 0.0468	0.9888 ± 0.0455
	ML-JNMF	$\mathbf{1.0000 \pm 0.0000}$	$\mathbf{0.0000 \pm 0.0000}$	$\mathbf{1.0000 \pm 0.0000}$	$\mathbf{1.0000 \pm 0.0000}$	$\mathbf{1.0000 \pm 0.0000}$
GNMLN 4	BLSC	0.3856 ± 0.2463	0.5544 ± 0.1122	0.0448 ± 0.0205	0.0246 ± 0.0114	0.7372 ± 0.4002
$n = 256$	GenLov	0.2742 ± 0.0805	0.5352 ± 0.0602	0.2735 ± 0.0806	0.2686 ± 0.0803	0.4305 ± 0.0853
$z_{out} = 8$	SymNMF	0.2254 ± 0.0927	0.4841 ± 0.0567	0.2208 ± 0.0872	0.1988 ± 0.1061	0.3522 ± 0.1204
$a = 0.0$	ONMTF	0.1665 ± 0.0356	0.6557 ± 0.0284	0.1551 ± 0.0326	0.1579 ± 0.0331	0.3313 ± 0.0359
	CSNMF	0.6517 ± 0.0662	0.2190 ± 0.0425	0.6517 ± 0.0662	0.5457 ± 0.0542	0.4652 ± 0.0182
	CPNMF	0.5962 ± 0.1017	0.2437 ± 0.0528	0.5962 ± 0.1017	0.4864 ± 0.0979	0.4332 ± 0.0536
	CSNMTF	0.4252 ± 0.1339	0.3380 ± 0.0671	0.4252 ± 0.1339	0.3455 ± 0.1333	0.3544 ± 0.0821
	GDG	0.2134 ± 0.0523	0.4900 ± 0.0363	0.2130 ± 0.0523	0.1795 ± 0.0525	0.3365 ± 0.0568
	SSCF	0.0367 ± 0.0220	0.6083 ± 0.0719	0.0362 ± 0.0211	0.0325 ± 0.0218	0.1897 ± 0.0254
	MTCP	0.3630 ± 0.0911	0.3684 ± 0.0463	0.3630 ± 0.0911	0.2850 ± 0.0899	0.3319 ± 0.0571
	ML-JNMF	$\mathbf{0.6556 \pm 0.0570}$	$\mathbf{0.2585 \pm 0.0480}$	$\mathbf{0.6354 \pm 0.0808}$	$\mathbf{0.6081 \pm 0.0982}$	$\mathbf{0.6987 \pm 0.1024}$
GNMLN 5	BLSC	0.6570 ± 0.2695	0.3080 ± 0.3230	0.5338 ± 0.4121	0.5226 ± 0.4215	0.9190 ± 0.1779
$n = 256$	GenLov	$\mathbf{0.9382 \pm 0.0347}$	$\mathbf{0.0309 \pm 0.0174}$	$\mathbf{0.9382 \pm 0.0347}$	$\mathbf{0.9381 \pm 0.0348}$	$\mathbf{0.9822 \pm 0.0109}$
$z_{out} = 8$	SymNMF	0.6103 ± 0.1568	0.1797 ± 0.0693	0.6103 ± 0.1568	0.5780 ± 0.1750	0.7332 ± 0.1426
$a = 1.0$	ONMTF	0.4730 ± 0.1213	0.2822 ± 0.0888	0.4450 ± 0.1413	0.4405 ± 0.1458	0.6719 ± 0.1089
	CSNMF	0.8454 ± 0.0979	0.0771 ± 0.0478	0.8454 ± 0.0979	0.8462 ± 0.1020	0.9383 ± 0.0589
	CPNMF	0.7818 ± 0.1218	0.1052 ± 0.0549	0.7818 ± 0.1218	0.7704 ± 0.1404	0.8932 ± 0.0970
	CSNMTF	0.5350 ± 0.1296	0.2336 ± 0.0438	0.5024 ± 0.1220	0.4981 ± 0.1513	0.7510 ± 0.1522
	GDG	0.5876 ± 0.1275	0.1856 ± 0.0496	0.5422 ± 0.1239	0.4752 ± 0.1351	0.6943 ± 0.1357
	SSCF	0.0171 ± 0.0102	0.5024 ± 0.0847	0.0131 ± 0.0073	0.0127 ± 0.0072	0.3104 ± 0.0240
	MTCP	0.4920 ± 0.1214	0.2427 ± 0.0498	0.4914 ± 0.1205	0.4782 ± 0.1398	0.7117 ± 0.1157
	ML-JNMF	0.9063 ± 0.0385	0.0472 ± 0.0195	0.9064 ± 0.0384	0.9109 ± 0.0374	0.9721 ± 0.0132

inter-layer graph encodes the phone calls interactions. More details about the constructed networks can be found in [29] and they are publicly available along with their annotated ground truth. In the MIT[11] network, the ground truth communities are the self-reported affiliations of the users, while in the NRC[12] network, the ground truth communities are the users' email affiliations.

The proposed approach is applied to both MIT and NRC networks to detect their community structure. The performance of the proposed approach is compared to other existing

[11] https://github.com/VGligorijevic/NF-CCE/tree/master/data/nets

[12] https://bitbucket.org/uuinfolab/20csur/src/master/algorithms/spectral/data/

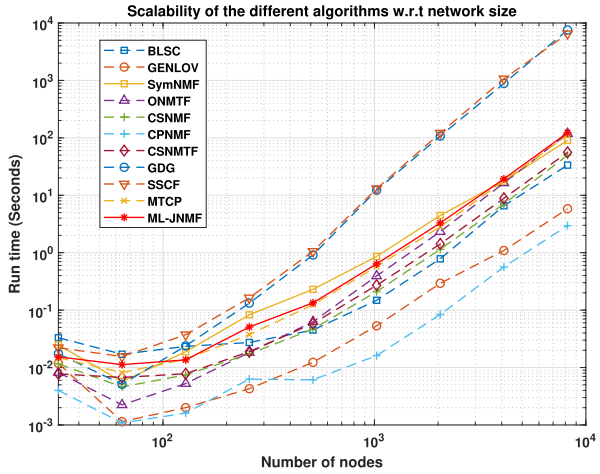

FIGURE 2. Run time vs. network size for the proposed method (ML-JNMF), block spectral clustering (BLSC), generalized louvain (GenLov), symmetric nonnegative matrix factorization (SymNMF), orthogonal nonnegative matrix tri-factorization (ONMTF), collective symmetric nonnegative matrix factorization (CSNMF), collective projective nonnegative matrix factorization (CSNMF), collective symmetric nonnegative matrix tri-factorization (CSNMTF), Geodesic Density Gradient (GDG) algorithm, subspace based community detection with fusion (SSCF) and multiplex cellular communities for tissue phenotyping (MCTP).

algorithms using different quality metrics and the results are reported in Table 6. The detected community structure is evaluated with respect to each one of the layers. As it can be seen in the table, the proposed approach outperforms the other algorithms with respect to multiple quality metrics in both layers.

2) SOCIAL NETWORKS

To evaluate the performance of the proposed algorithm in detecting the community structure in real social multi-layer networks, a multi-layer network corresponding to the relationships between workers at Lazega law firm[13] (LLF) [68], [69] is adopted. This network models the interactions between 71 partners and associates who work at a corporate law partnership. The network has three layers, where each layer encodes a particular relationship, i.e., *co-work*, *advice* and *friendship*. In addition, the data set includes 7 attributes that can be used to evaluate the detected community structure. In this paper, the (1) status (partner or associate) (LLF_S), and (3) office location (Boston, Hartford, or Providence) (LLF_O) are used as the ground truth for the communities.

A homogeneous 2-layer, unweighted and undirected network is constructed where the intra-layer graphs encode work relations, i.e., *co-work* and *advice*, whereas the inter-layer graph models the *friendship* relations.

The proposed algorithm is applied to the constructed network and the office and status attributes are used as the ground truth to evaluate the detected community structure. Table 6 presents a comparison between the different methods in terms of NMI, VI, F-value, recall and purity of the detected community structure with respect to each of the ground truth attributes. Since both layers consist of the same set of nodes, the detected community structure is evaluated with respect to each one of the layers. As it can be seen in Table 6, for both layers, the detected community structure is closely related to the office locations, which agrees with previous studies [70], [71]. We also note that layer 2, which encodes the *co-work* interactions among the workers, reflects the community structure of the network with respect to the office location better than layer 1, which encodes the *advice* interactions. On the other hand, the community structure detected from the *advice* layer is more closely related to the status (partner or associate) than the *co-work* layer.

3) BIOLOGICAL DATA: C. ELEGANS NETWORK

The C. Elegans[14] network [3], [72] is a multi-layer network that consists of 279 nodes. The edges in the network reflect the different synaptic junctions, namely electric, chemical monadic, and polyadic between the neurons of the Caenorhabditis Elegans connectome. This data set, also, includes three attributes that can be used as the ground truth communities. In this paper, (1) the group of the neuron (body-wall, mechanosensory, head motor neurons, etc.) (CEleg_G), and (2) the color of the neuron (grey, red, orange, etc.) (CEleg_C) are used to evaluate the ability of the different algorithms in detecting the community structure. The performance comparison between the different methods is given in Table 6. As it can be seen from the table, the proposed method can detect the community structure better than the other methods with respect to NMI, purity and recall metrics compared to the group and color attributes.

4) HAND-WRITTEN DIGITS DATA: UCI

The UCI[15] [73] data set consists of features of handwritten digits from (0- 9) extracted from a collection of Dutch utility maps. There are 200 patterns per digit which results in a total of 2000 patterns that have been digitized in binary images. These digits are represented by different feature sets including (1) Fourier coefficients of the character shapes, (2) profile correlations and (3) Karhunen-Loéve coefficients. Intra- and inter-layer graphs, $\mathbf{A}^1, \mathbf{A}^2 \in \mathbb{R}^{2000\times2000}$ and $\mathbf{A}^{12} \in \mathbb{R}^{2000\times2000}$, are constructed using the Guassian kernel similarity function [48] and keeping the nearest 100 neighbors. The different algorithms are applied to the constructed network and their performance comparison is given in Table 7. The proposed approach performs better than all the other methods in terms of all quality metrics in the second layer and in terms of purity in the first layer.

5) CALTECH101 DATA SET

The Caltech101[16] [74] data set is a well-known object recognition data set. It contains 102 objects, i.e., communities, with $40 - 800$ images for each object which results in 9144 images in total. The original data set consists of 6 types

[13] https://manliodedomenico.com/data.php

[14] https://manliodedomenico.com/data.php

[15] https://archive.ics.uci.edu/ml/datasets/Multiple+Features

[16] http://www.vision.caltech.edu/Image_Datasets/Caltech101/Caltech101.html

TABLE 5. Performance comparison between the different methods in detecting the community structure as the network's size increases.

Network	Method	NMI	VI	F-value	Recall	Purity
Network 1 $n = 32$	BLSC	0.9333 ± 0.1155	0.0333 ± 0.0577	0.9333 ± 0.1155	0.8889 ± 0.1925	0.9167 ± 0.1443
	GenLov	0.9333 ± 0.1155	0.0333 ± 0.0577	0.9333 ± 0.1155	0.8889 ± 0.1925	0.9167 ± 0.1443
	SymNMF	0.6238 ± 0.0764	0.2272 ± 0.0491	0.6238 ± 0.0764	0.6264 ± 0.0697	0.7500 ± 0.0000
	ONMTF	0.7786 ± 0.1917	0.1326 ± 0.1148	0.7786 ± 0.1917	0.7778 ± 0.1925	0.8333 ± 0.1443
	CSNMF	0.8000 ± 0.0000	0.1000 ± 0.0000	0.8000 ± 0.0000	0.6667 ± 0.0000	0.7500 ± 0.0000
	CPNMF	0.5520 ± 0.2361	0.2687 ± 0.1517	0.5520 ± 0.2361	0.5400 ± 0.2194	0.6875 ± 0.1083
	CSNMTF	0.6811 ± 0.0125	0.1874 ± 0.0109	0.6811 ± 0.0125	0.6667 ± 0.0000	0.7500 ± 0.0000
	GDG	0.0863 ± 0.0198	0.3851 ± 0.0037	0.0863 ± 0.0198	0.0609 ± 0.0155	0.5208 ± 0.0180
	SSCF	0.0987 ± 0.0246	0.4358 ± 0.0325	0.0987 ± 0.0246	0.0793 ± 0.0202	0.5208 ± 0.0180
	MTCP	0.8000 ± 0.0000	0.1000 ± 0.0000	0.8000 ± 0.0000	0.6667 ± 0.0000	0.7500 ± 0.0000
	ML-JNMF	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
Network2 $n = 64$	BLSC	0.9333 ± 0.1155	0.0278 ± 0.0481	0.9333 ± 0.1155	0.8889 ± 0.1925	0.9167 ± 0.1443
	GenLov	0.9333 ± 0.1155	0.0278 ± 0.0481	0.9333 ± 0.1155	0.8889 ± 0.1925	0.9167 ± 0.1443
	SymNMF	0.7272 ± 0.2483	0.1352 ± 0.1226	0.7272 ± 0.2483	0.7245 ± 0.2516	0.8333 ± 0.1443
	ONMTF	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	CSNMF	0.6710 ± 0.0065	0.1635 ± 0.0048	0.6710 ± 0.0065	0.6667 ± 0.0000	0.7500 ± 0.0000
	CPNMF	0.5542 ± 0.2149	0.2238 ± 0.1136	0.5542 ± 0.2149	0.5488 ± 0.2041	0.7031 ± 0.0812
	CSNMTF	0.6675 ± 0.0007	0.1660 ± 0.0005	0.6675 ± 0.0007	0.6667 ± 0.0000	0.7500 ± 0.0000
	GDG	0.0275 ± 0.0072	0.4206 ± 0.0320	0.0275 ± 0.0072	0.0240 ± 0.0081	0.5052 ± 0.0090
	SSCF	0.0329 ± 0.0180	0.4390 ± 0.0377	0.0329 ± 0.0180	0.0299 ± 0.0156	0.5104 ± 0.0090
	MTCP	0.8000 ± 0.0000	0.0833 ± 0.0000	0.8000 ± 0.0000	0.6667 ± 0.0000	0.7500 ± 0.0000
	ML-JNMF	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
Network3 $n = 128$	BLSC	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	GenLov	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	SymNMF	0.8326 ± 0.1833	0.0720 ± 0.0787	0.8326 ± 0.1833	0.8352 ± 0.1830	0.9036 ± 0.1345
	ONMTF	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	CSNMF	0.6846 ± 0.0232	0.1320 ± 0.0139	0.6846 ± 0.0232	0.6667 ± 0.0000	0.7500 ± 0.0000
	CPNMF	0.5552 ± 0.0997	0.1910 ± 0.0432	0.5552 ± 0.0997	0.5559 ± 0.0982	0.7240 ± 0.0274
	CSNMTF	0.6727 ± 0.0053	0.1390 ± 0.0033	0.6727 ± 0.0053	0.6667 ± 0.0000	0.7500 ± 0.0000
	GDG	0.0105 ± 0.0073	0.3867 ± 0.0353	0.0105 ± 0.0073	0.0098 ± 0.0072	0.5000 ± 0.0000
	SSCF	0.0060 ± 0.0018	0.4233 ± 0.0131	0.0060 ± 0.0018	0.0060 ± 0.0019	0.5000 ± 0.0000
	MTCP	0.8000 ± 0.0000	0.0714 ± 0.0000	0.8000 ± 0.0000	0.6667 ± 0.0000	0.7500 ± 0.0000
	ML-JNMF	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
Network4 $n = 256$	BLSC	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	GenLov	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	SymNMF	0.7527 ± 0.2142	0.0934 ± 0.0809	0.7527 ± 0.2142	0.7561 ± 0.2118	0.8529 ± 0.1307
	ONMTF	0.7826 ± 0.1884	0.0807 ± 0.0699	0.7826 ± 0.1884	0.7778 ± 0.1925	0.8333 ± 0.1443
	CSNMF	0.6748 ± 0.0081	0.1205 ± 0.0045	0.6748 ± 0.0081	0.6667 ± 0.0000	0.7500 ± 0.0000
	CPNMF	0.5378 ± 0.2440	0.1714 ± 0.0919	0.5378 ± 0.2440	0.5293 ± 0.2378	0.7096 ± 0.0699
	CSNMTF	0.6910 ± 0.0126	0.1119 ± 0.0067	0.6910 ± 0.0126	0.6667 ± 0.0000	0.7500 ± 0.0000
	GDG	0.0063 ± 0.0027	0.3692 ± 0.0036	0.0063 ± 0.0027	0.0062 ± 0.0027	0.5000 ± 0.0000
	SSCF	0.0086 ± 0.0067	0.3769 ± 0.0064	0.0086 ± 0.0067	0.0086 ± 0.0067	0.5000 ± 0.0000
	MTCP	0.8000 ± 0.0000	0.0625 ± 0.0000	0.8000 ± 0.0000	0.6667 ± 0.0000	0.7500 ± 0.0000
	ML-JNMF	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
Network5 $n = 512$	BLSC	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	GenLov	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	SymNMF	0.8129 ± 0.1732	0.0628 ± 0.0578	0.8129 ± 0.1732	0.8175 ± 0.1725	0.8880 ± 0.1270
	ONMTF	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	CSNMF	0.6908 ± 0.0263	0.0998 ± 0.0122	0.6908 ± 0.0263	0.6667 ± 0.0000	0.7500 ± 0.0000
	CPNMF	0.2316 ± 0.1130	0.2617 ± 0.0379	0.2316 ± 0.1130	0.2368 ± 0.1160	0.6035 ± 0.0380
	CSNMTF	0.6684 ± 0.0012	0.1102 ± 0.0006	0.6684 ± 0.0012	0.6667 ± 0.0000	0.7500 ± 0.0000
	GDG	0.0011 ± 0.0009	0.3357 ± 0.0056	0.0011 ± 0.0009	0.0011 ± 0.0010	0.5000 ± 0.0000
	SSCF	0.0040 ± 0.0014	0.3370 ± 0.0066	0.0040 ± 0.0014	0.0041 ± 0.0014	0.5000 ± 0.0000
	MTCP	0.8000 ± 0.0000	0.0556 ± 0.0000	0.8000 ± 0.0000	0.6667 ± 0.0000	0.7500 ± 0.0000
	ML-JNMF	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
Network6 $n = 1024$	BLSC	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	GenLov	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	SymNMF	0.7176 ± 0.2364	0.0852 ± 0.0716	0.7176 ± 0.2364	0.7198 ± 0.2350	0.8317 ± 0.1415
	ONMTF	0.8207 ± 0.1652	0.0501 ± 0.0483	0.8207 ± 0.1652	0.7778 ± 0.1925	0.8333 ± 0.1443
	CSNMF	0.6678 ± 0.0013	0.0995 ± 0.0006	0.6678 ± 0.0013	0.6667 ± 0.0000	0.7500 ± 0.0000
	CPNMF	0.1316 ± 0.0629	0.2650 ± 0.0201	0.1316 ± 0.0629	0.1337 ± 0.0635	0.5739 ± 0.0458
	CSNMTF	0.6747 ± 0.0069	0.0965 ± 0.0031	0.6747 ± 0.0069	0.6667 ± 0.0000	0.7500 ± 0.0000
	GDG	0.0007 ± 0.0005	0.3063 ± 0.0009	0.0007 ± 0.0005	0.0007 ± 0.0006	0.5000 ± 0.0000
	SSCF	0.0012 ± 0.0007	0.3033 ± 0.0011	0.0012 ± 0.0007	0.0012 ± 0.0007	0.5000 ± 0.0000
	MTCP	0.8000 ± 0.0000	0.0500 ± 0.0000	0.8000 ± 0.0000	0.6667 ± 0.0000	0.7500 ± 0.0000
	ML-JNMF	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
Network7 $n = 2048$	BLSC	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	GenLov	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	SymNMF	0.6236 ± 0.0372	0.1031 ± 0.0107	0.6236 ± 0.0372	0.6263 ± 0.0342	0.7500 ± 0.0000
	ONMTF	0.9191 ± 0.1401	0.0194 ± 0.0336	0.9191 ± 0.1401	0.8889 ± 0.1925	0.9167 ± 0.1443
	CSNMF	0.6712 ± 0.0039	0.0891 ± 0.0016	0.6712 ± 0.0039	0.6667 ± 0.0000	0.7500 ± 0.0000
	CPNMF	0.2328 ± 0.1998	0.2126 ± 0.0572	0.2328 ± 0.1998	0.2348 ± 0.1994	0.6034 ± 0.0977
	CSNMTF	0.6853 ± 0.0038	0.0835 ± 0.0015	0.6853 ± 0.0038	0.6667 ± 0.0000	0.7500 ± 0.0000
	GDG	0.0003 ± 0.0001	0.2766 ± 0.0053	0.0003 ± 0.0001	0.0003 ± 0.0001	0.5000 ± 0.0000
	SSCF	0.0014 ± 0.0002	0.2791 ± 0.0007	0.0014 ± 0.0002	0.0014 ± 0.0002	0.5000 ± 0.0000
	MTCP	0.8000 ± 0.0000	0.0455 ± 0.0000	0.8000 ± 0.0000	0.6667 ± 0.0000	0.7500 ± 0.0000
	ML-JNMF	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
Network8 $n = 4096$	BLSC	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	GenLov	0.9333 ± 0.1155	0.0139 ± 0.0241	0.9333 ± 0.1155	0.8889 ± 0.1925	0.9167 ± 0.1443
	SymNMF	0.6802 ± 0.0177	0.0785 ± 0.0063	0.6802 ± 0.0177	0.6667 ± 0.0000	0.7500 ± 0.0000
	ONMTF	0.8201 ± 0.1682	0.0422 ± 0.0417	0.8201 ± 0.1682	0.7778 ± 0.1925	0.8333 ± 0.1443
	CSNMF	0.6722 ± 0.0090	0.0813 ± 0.0033	0.6722 ± 0.0090	0.6667 ± 0.0000	0.7500 ± 0.0000
	CPNMF	0.1733 ± 0.0270	0.2102 ± 0.0071	0.1733 ± 0.0270	0.1762 ± 0.0273	0.5677 ± 0.0218
	CSNMTF	0.6836 ± 0.0222	0.0773 ± 0.0078	0.6836 ± 0.0222	0.6667 ± 0.0000	0.7500 ± 0.0000
	GDG	0.0004 ± 0.0003	0.2535 ± 0.0043	0.0004 ± 0.0003	0.0004 ± 0.0003	0.5000 ± 0.0000
	SSCF	0.0004 ± 0.0001	0.2569 ± 0.0001	0.0004 ± 0.0001	0.0004 ± 0.0002	0.5000 ± 0.0000
	MTCP	0.8000 ± 0.0000	0.0417 ± 0.0000	0.8000 ± 0.0000	0.6667 ± 0.0000	0.7500 ± 0.0000
	ML-JNMF	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
Network9 $n = 8192$	BLSC	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	GenLov	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**
	SymNMF	0.7278 ± 0.2400	0.0621 ± 0.0545	0.7278 ± 0.2400	0.7243 ± 0.2446	0.8212 ± 0.1547
	ONMTF	0.8889 ± 0.1925	0.0256 ± 0.0444	0.8889 ± 0.1925	0.8889 ± 0.1925	0.9167 ± 0.1443
	CSNMF	0.6892 ± 0.0222	0.0695 ± 0.0072	0.6892 ± 0.0222	0.6667 ± 0.0000	0.7500 ± 0.0000
	CPNMF	0.1464 ± 0.0955	0.2004 ± 0.0236	0.1464 ± 0.0955	0.1485 ± 0.0967	0.5699 ± 0.0578
	CSNMTF	0.7005 ± 0.0351	0.0662 ± 0.0109	0.7005 ± 0.0351	0.6667 ± 0.0000	0.7500 ± 0.0000
	GDG	0.0001 ± 0.0001	0.2358 ± 0.0023	0.0001 ± 0.0001	0.0001 ± 0.0001	0.5000 ± 0.0000
	SSCF	0.0003 ± 0.0001	0.2371 ± 0.0001	0.0003 ± 0.0001	0.0003 ± 0.0001	0.5000 ± 0.0000
	MTCP	0.8000 ± 0.0000	0.0385 ± 0.0000	0.8000 ± 0.0000	0.6667 ± 0.0000	0.7500 ± 0.0000
	ML-JNMF	**1.0000 ± 0.0000**	**0.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**	**1.0000 ± 0.0000**

TABLE 6. Performance comparison between the proposed method (ML-JNMF), block spectral clustering (BLSC), generalized louvain (GenLov), symmetric nonnegative matrix factorization (SymNMF), orthogonal nonnegative matrix tri-factorization (ONMTF), collective symmetric nonnegative matrix factorization (CSNMF), collective projective nonnegative matrix factorization (CSNMF), collective symmetric nonnegative matrix tri-factorization (CSNMTF), Geodesic Density Gradient (GDG) algorithm, subspace based community detection with fusion (SSCF) and multiplex cellular communities for tissue phenotyping (MCTP) in detecting the community structure in real multi-layer networks. The comparison is conducted in terms of average NMI, VI, F-value, recall and purity. The best performance is shown in bold and when ML-JNMF achieves the second best performance is underlined.

	Metric	BLSC	GenLov	SymNMF	ONMTF	CSNMF	CPNMF	CSNMTF	GDG	SSCF	MTCP	ML-JNMF
MIT, L1	NMI	0.1180	0.1808	0.0668	0.2406	0.3186	0.3462	0.3343	0.1530	0.1145	0.2788	**0.4223**
	VI	0.4362	0.4990	0.4402	0.5907	0.3312	0.4565	**0.3227**	0.4674	0.7023	0.3496	0.4650
	F-value	0.1180	0.1592	0.0668	0.1836	0.3186	**0.3462**	0.3343	0.1530	0.0884	0.2788	0.3069
	Recall	0.0865	0.1420	0.0467	0.1976	0.2295	**0.3582**	0.2401	0.1251	0.0952	0.2003	0.3077
	Purity	0.4828	0.4943	0.4598	0.5632	0.6207	0.6437	0.6207	0.4762	0.4253	0.5977	**0.6667**
MIT, L2	NMI	0.4613	**0.5251**	0.3441	0.3821	0.3186	0.3462	0.3343	0.4446	0.1516	0.2788	0.4615
	VI	0.3713	**0.3062**	0.3188	0.4496	0.3312	0.4565	0.3227	0.3301	0.6722	0.3496	0.3661
	F-value	0.4613	0.3736	0.3441	0.3849	0.3186	0.3462	0.3343	0.4446	0.1017	0.2788	**0.4693**
	Recall	0.3530	0.3146	0.2478	0.3886	0.2295	0.3582	0.2401	0.4053	0.1116	0.2003	**0.4106**
	Purity	0.6437	0.6897	0.6092	0.6437	0.6207	0.6437	0.6207	0.6707	0.4713	0.5977	**0.6782**
NRC, L1	NMI	0.0941	0.1650	0.0271	0.0274	0.3338	0.1591	0.2890	0.0843	0.0942	0.2496	**0.3749**
	VI	**0.3836**	0.4928	0.4162	0.4274	0.4236	0.5739	0.4091	0.4174	0.6928	0.4341	0.4903
	F-value	0.0941	0.1650	0.0271	0.0274	0.3338	0.1591	0.2890	0.0843	0.0942	0.2496	**0.3442**
	Recall	0.0558	0.1363	0.0162	0.0169	0.2972	0.1520	0.2328	0.0538	0.1009	0.2022	**0.3553**
	Purity	0.3162	0.3824	0.2868	0.3015	0.5294	0.4044	0.4706	0.2941	0.3750	0.4412	**0.5588**
NRC, L2	NMI	0.1079	0.3692	0.0645	0.0657	0.3338	0.1591	0.2890	0.3301	0.1335	0.2496	**0.4180**
	VI	0.3912	**0.3813**	0.3972	0.3871	0.4236	0.5739	0.4091	0.3990	0.6370	0.4341	0.4676
	F-value	0.1079	0.3692	0.0645	0.0657	0.3338	0.1591	0.2890	0.3301	0.1335	0.2496	**0.4064**
	Recall	0.0662	0.3124	0.0383	0.0381	0.2972	0.1520	0.2328	0.2753	0.1374	0.2022	**0.4487**
	Purity	0.3162	0.4779	0.3088	0.3162	0.5294	0.4044	0.4706	0.4779	0.3235	0.4412	**0.6103**
LLF_O, L1	NMI	0.4705	0.5816	0.5760	0.5666	0.5732	0.2725	0.5760	0.4694	0.0495	0.5732	**0.7054**
	VI	0.2344	0.1818	0.1858	0.1891	0.1870	0.3952	0.1858	0.2277	0.4310	0.1870	**0.1157**
	F-value	0.4705	0.5816	0.5760	0.5666	0.5732	0.2876	0.5760	0.4694	0.0495	0.5732	**0.7054**
	Recall	0.5695	0.6911	0.6900	0.6757	0.6869	0.3828	0.6900	0.5508	0.0613	0.6869	**0.7578**
	Purity	0.8873	0.9296	0.9296	0.9296	0.9296	0.6761	0.9296	0.8732	0.6761	0.9296	**0.9437**
LLF_O, L2	NMI	0.5168	0.5816	0.5760	0.5666	0.5732	0.2725	0.5760	0.6102	0.0160	0.5732	**0.7811**
	VI	0.2085	0.1818	0.1858	0.1891	0.1870	0.3952	0.1858	0.1703	0.4104	0.1870	**0.0741**
	F-value	0.5168	0.5816	0.5760	0.5666	0.5732	0.2876	0.5760	0.6102	0.0160	0.5732	**0.7811**
	Recall	0.6099	0.6911	0.6900	0.6757	0.6869	0.3828	0.6900	0.7287	0.0182	0.6869	**0.7232**
	Purity	0.9014	0.9296	0.9296	0.9296	0.9296	0.8310	0.9296	0.9296	0.6761	0.9296	**0.9437**
LLF_S, L1	NMI	0.2131	0.0420	0.0652	0.1509	0.0418	0.0569	0.0478	0.0807	0.0320	0.0569	**0.3027**
	VI	0.3324	0.3969	0.3906	0.3531	0.4005	0.4491	0.3978	0.3759	0.4193	0.3942	**0.3302**
	F-value	0.1275	0.0420	0.0652	**0.1509**	0.0418	0.0341	0.0478	0.0807	0.0268	0.0569	0.1011
	Recall	0.1332	0.0535	0.0838	**0.1930**	0.0537	0.0422	0.0614	0.1015	0.0332	0.0731	0.1140
	Purity	0.7746	0.6056	0.6620	0.7324	0.6197	0.8310	0.6338	0.6761	0.6197	0.6479	**0.8169**
LLF_S, L2	NMI	0.1685	0.0420	0.1008	0.1830	0.0418	0.0569	0.0478	0.1005	0.0816	0.0569	**0.2851**
	VI	0.3420	0.3969	0.3759	0.3410	0.4005	0.4491	0.3978	0.3747	0.3644	0.3942	**0.3307**
	F-value	0.0789	0.0420	0.1008	**0.1830**	0.0418	0.0341	0.0478	0.1005	0.0020	0.0569	0.0750
	Recall	0.0787	0.0535	0.1296	**0.2350**	0.0537	0.0422	0.0614	0.1287	0.0024	0.0731	0.0827
	Purity	0.7465	0.6056	0.7042	0.7606	0.6197	0.6479	0.6338	0.7042	0.5634	0.6479	**0.8028**
$CEleg_G$, L1	NMI	**0.4237**	0.4118	0.3483	0.3527	0.3259	0.2467	0.2677	0.4034	0.0453	0.3459	0.4142
	VI	**0.3509**	0.4033	0.4135	0.5103	0.4152	0.4379	0.4111	0.3942	0.6136	0.4142	0.4536
	F-value	**0.4237**	0.4118	0.3483	0.3527	0.3259	0.2647	0.2677	0.4034	0.0453	0.3459	0.4142
	Recall	0.3330	0.3643	0.2852	0.3589	0.2591	0.2034	0.1939	0.3440	0.0376	0.2826	**0.4139**
	Purity	0.4387	0.4461	0.4164	0.4349	0.4052	0.3346	0.3383	0.4387	0.2639	0.4126	**0.4610**
$CEleg_G$, L2	NMI	0.4221	0.4449	0.3976	0.3764	0.3259	0.2467	0.2677	0.4302	0.0481	0.3459	**0.4831**
	VI	0.3491	0.3785	0.3808	0.4842	0.4152	0.4379	0.4111	**0.3641**	0.6284	0.4142	0.4328
	F-value	0.4221	**0.4449**	0.3976	0.3764	0.3259	0.2647	0.2677	0.4302	0.0481	0.3459	0.4273
	Recall	0.3290	0.3914	0.3243	0.3771	0.2591	0.2034	0.1939	0.3547	0.0410	0.2826	**0.4165**
	Purity	0.4387	0.4721	0.4238	0.4387	0.4052	0.3346	0.3383	0.4498	0.2639	0.4126	**0.5390**
$CEleg_C$, L1	NMI	0.3867	0.4148	0.1075	0.3675	0.3897	0.3394	0.3284	0.3325	0.0214	0.2111	**0.4661**
	VI	**0.3977**	0.4085	0.5085	0.5828	0.4460	0.5108	0.4517	0.4351	0.5965	0.4344	0.5099
	F-value	0.3867	**0.4148**	0.1075	0.3514	0.3897	0.3392	0.3284	0.3325	0.0214	0.2111	0.4137
	Recall	0.2936	0.3391	0.0717	0.3650	0.3335	0.3070	0.2587	0.2538	0.0153	0.1361	**0.4290**
	Purity	0.4164	0.4089	0.2751	0.4535	0.4461	0.4275	0.4089	0.4201	0.2639	0.3383	**0.5279**
$CEleg_C$, L2	NMI	0.3854	0.4206	0.1622	0.4205	0.3897	0.3394	0.3284	0.3780	0.0469	0.2111	**0.4767**
	VI	0.3955	0.4007	0.4560	0.5398	0.4460	0.5108	0.4517	**0.3936**	0.5776	0.4344	0.4849
	F-value	0.3854	0.4206	0.1622	0.3849	0.3897	0.3392	0.3284	0.3780	0.0469	0.2111	**0.4299**
	Recall	0.2904	0.3406	0.1033	0.4061	0.3335	0.3070	0.2587	0.2801	0.0333	0.1361	**0.4523**
	Purity	0.4164	0.4089	0.3048	0.5093	0.4461	0.4275	0.4089	0.4238	0.2639	0.3383	**0.5242**

TABLE 7. Performance comparison between the proposed method (ML-JNMF), block spectral clustering (BLSC), generalized louvain (GenLov), symmetric nonnegative matrix factorization (SymNMF), orthogonal nonnegative matrix tri-factorization (ONMTF), collective symmetric nonnegative matrix factorization (CSNMF), collective projective nonnegative matrix factorization (CSNMF), collective symmetric nonnegative matrix tri-factorization (CSNMTF), Geodesic Density Gradient (GDG) algorithm, subspace based community detection with fusion (SSCF) and multiplex cellular communities for tissue phenotyping (MCTP) in detecting the community structure in UCI and Caltech101 networks. The comparison is conducted in terms of average NMI, VI, F-value, recall and purity. The best performance is shown in bold and when ML-JNMF achieves the second best performance is underlined.

	Metric	BLSC	GenLov	SymNMF	ONMTF	CSNMF	CPNMF	CSNMTF	GDG	SSCF	MTCP	ML-JNMF
UCI, L1	NMI	0.7633	0.7880	0.4826	0.3342	0.7092	0.7085	**0.8134**	0.5985	0.0071	0.7164	0.8027
	VI	0.1483	0.1206	0.2936	0.2458	0.1559	0.1547	**0.1096**	0.2365	0.5334	0.1519	0.1159
	F-value	0.7286	0.7880	0.4826	0.3342	0.7092	0.7085	**0.8134**	0.5985	0.0071	0.7164	0.8027
	Recall	0.7330	0.7399	0.4520	0.2036	0.6278	0.6206	**0.7888**	0.5818	0.0063	0.6335	0.7781
	Purity	0.8160	0.7355	0.4760	0.2000	0.5670	0.5810	0.8220	0.6700	0.1310	0.5695	**0.8235**
UCI, L2	NMI	0.7741	0.8119	0.5018	0.3345	0.7092	0.7085	0.8134	0.7265	0.0059	0.7164	**0.8278**
	VI	0.1413	0.1071	0.2827	0.2456	0.1559	0.1547	0.1096	0.1589	0.5278	0.1519	**0.1012**
	F-value	0.7393	0.8119	0.5018	0.3345	0.7092	0.7085	0.8134	0.7265	0.0059	0.7164	**0.8278**
	Recall	0.7418	0.7628	0.4701	0.2037	0.6278	0.6206	0.7888	0.6965	0.0052	0.6335	**0.8026**
	Purity	0.8205	0.7475	0.4995	0.2000	0.5670	0.5810	0.8220	0.7020	0.1260	0.5695	**0.8345**
Caltech, L1	NMI	0.1536	0.1013	0.2276	0.2498	**0.3231**	0.1650	0.2791	0.2208	0.1290	0.2739	0.2880
	VI	0.5928	**0.5445**	0.6682	0.6712	0.6515	0.6752	0.6564	0.7235	0.8396	0.6397	0.6839
	F-value	0.1536	0.1013	0.2276	0.2498	**0.3231**	0.1650	0.2791	0.2208	0.1290	0.2739	0.2880
	Recall	0.1176	0.0671	0.2151	0.2442	**0.3398**	0.1458	0.2517	0.2240	0.1358	0.2637	0.2784
	Purity	0.2030	0.1663	0.2253	0.2504	**0.3040**	0.1810	0.2758	0.1964	0.1083	0.2652	0.2922
Caltech, L2	NMI	0.1981	0.1179	0.2363	0.2624	0.3231	0.1650	0.2891	0.0745	0.1315	0.2739	**0.3268**
	VI	0.5723	**0.5393**	0.6508	0.6455	0.6515	0.6752	0.6564	0.5453	0.8373	0.6397	0.6532
	F-value	0.1981	0.1179	0.2363	0.2624	0.3231	0.1650	0.2891	0.0745	0.1315	0.2739	**0.3262**
	Recall	0.1544	0.0788	0.2200	0.2509	**0.3398**	0.1458	0.2917	0.0479	0.1385	0.2637	0.3139
	Purity	0.2418	0.1856	0.2421	0.2603	0.3040	0.1810	0.2758	0.1153	0.1065	0.2652	**0.3102**

of features [75]. In our experiments we selected three types to construct the multi-layer network including (1) Gabor feature, (2) wavelet moments and (3) CENTRIST feature. Intra- and inter-layer graphs, $\mathbf{A}^1, \mathbf{A}^2 \in \mathbb{R}^{9144\times9144}$ and $\mathbf{A}^{12} \in \mathbb{R}^{9144\times9144}$, are constructed using the Guassian kernel similarity function [48] and keeping the nearest 1000 neighbors. Since this is a large data set, the number of communities is given as an input to all algorithms. The performance of the different methods in detecting the community structure of the Caltech101 network is compared and the results are reported in Table 7. As it can be seen from the table, the proposed algorithm performs better than the other algorithms in terms of NMI, F-value and purity in the second layer. However, the CSNMF approach preforms slightly better than the proposed approach in the first layer. The reduced accuracy of all the methods for this case is mainly due to the high inter-cluster edge density.

6) IMDB MULTI-LAYER NETWORK

To assess the ability of the proposed algorithm in detecting the community structure in heterogeneous multi-layer networks, a data set that contains a collection of best and worst movies from the Internet Movie DataBase (IMDB) is adopted. The complete IMDB data set is publicly available in SQL format.[17] In this paper, we used an IMDB data set consisting of movies on the IMDB's top 250 and bottom 100 chart along with their genres and directors with 166 movies, 130 directors and 20 movie and director genres.

a: HETEROGENEOUS MULTI-LAYER NETWORK CONSTRUCTION

A 2-layer multi-layer network is constructed to model the IMDB data set. The intra-layer graphs model the relations among directors and movies, respectively. On the other hand, the inter-layer graph models the relation between directors and movies. Intra-layer adjacency matrices are constructed based on genre similarity. Genre can be considered as a categorical variable as it can only take on a fixed number of values. For example, in the first layer, a pair of nodes, i.e., directors, are connected if they directed the same genre, where each director can direct the same genre multiple times. Consequently, for every director, a binary-valued vector, $g \in \mathbb{R}^{1\times20}$, is generated where the entries of the vector are 1 if the director directed a movie in that genre and 0 otherwise. Pair-wise Pearson correlation coefficient is then calculated between the directors' genre vectors. The final adjacency matrix is constructed by keeping the edges with correlation values greater than 0.5. The second layer corresponds to the movies and is constructed based on the genre similarity following the procedure described above. The inter-layer adjacency matrix that models director-movie relations is constructed such that a node from the director layer is connected to a node from the movie layer if the director directed that movie. The constructed network and the IMDB data set details are available in our GitHub repository.[18]

[17] https://relational.fit.cvut.cz/dataset/IMDb

[18] https://github.com/EsraaAlsharoa/IMDB-DATA-SET.git

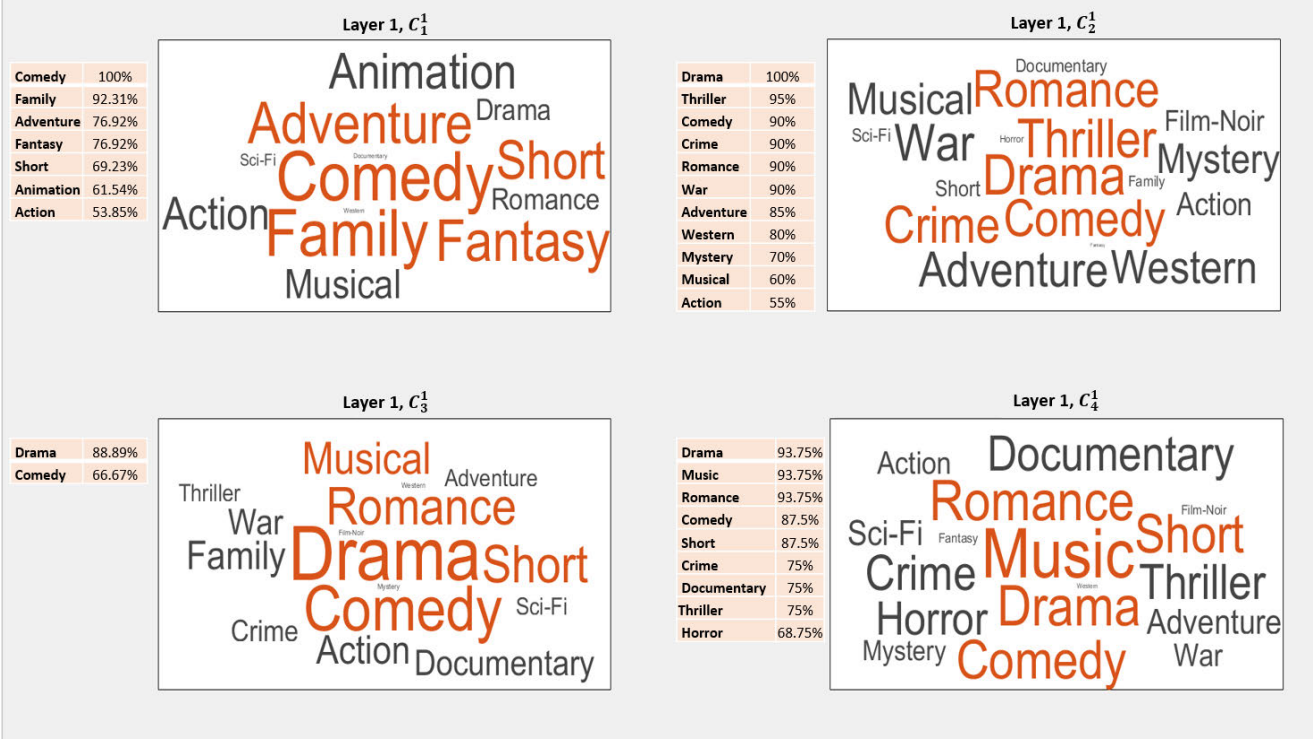

FIGURE 3. The detected set of intra-layer communities in the directors layer of the IMDB multi-layer network.

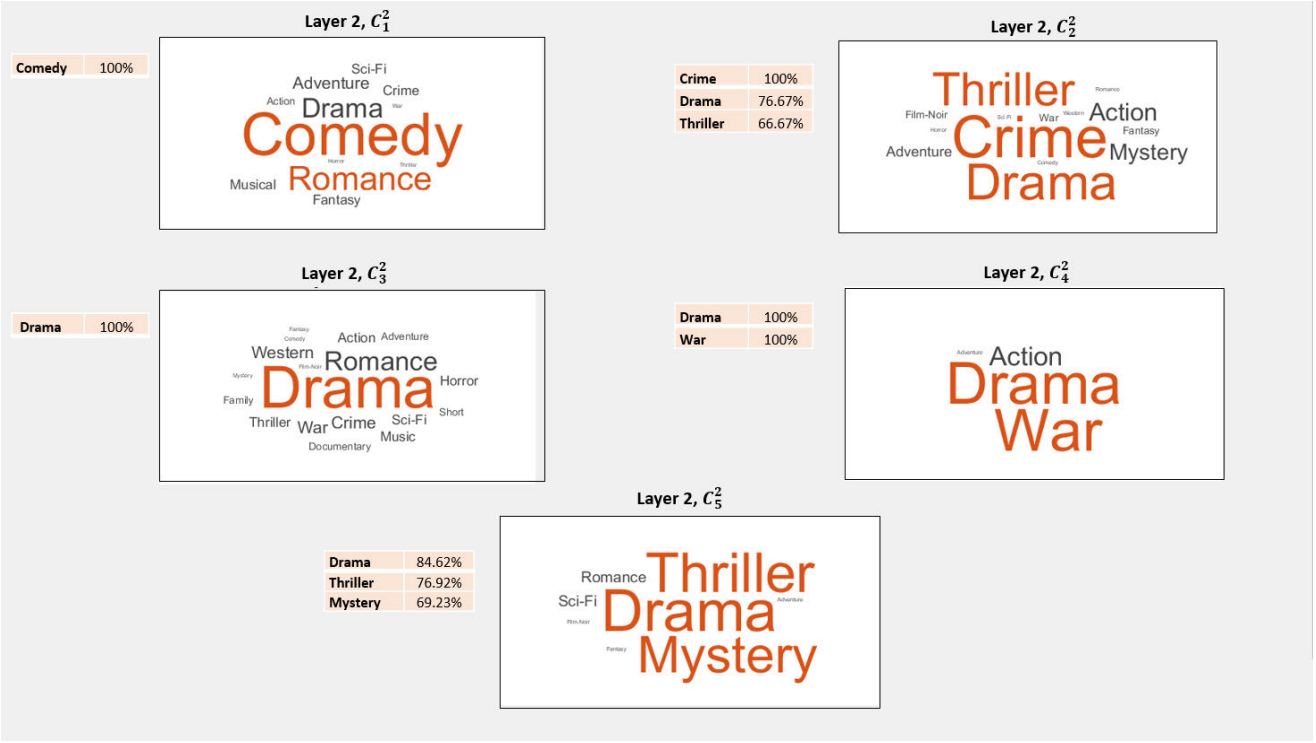

FIGURE 4. The detected set of intra-layer communities in the movies layer of the IMDB multi-layer network.

The proposed approach is applied to detect the community structure of the IMDB multi-layer network. The proposed algorithm detects 4 intra-layer communities in the directors layer and 5 intra-layer communities in the movies layer as well as 2 inter-layer communities. In order to analyze and interpret the detected community structure, the shared genres

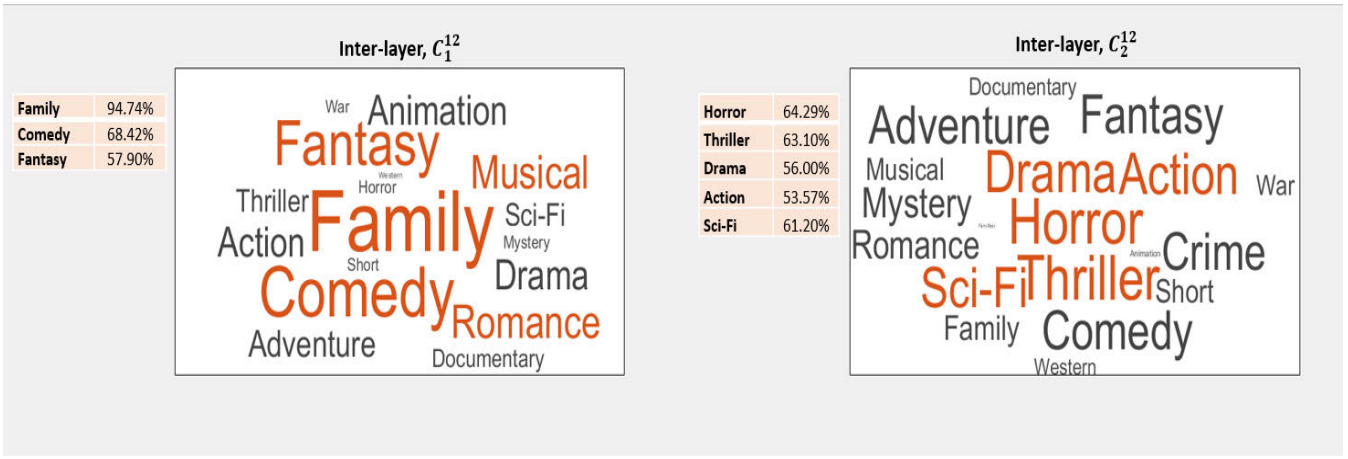

FIGURE 5. The detected set of inter-layer communities in the directors-movies inter-layer of the IMDB multi-layer network.

between the nodes in the directors, movies and directors-movies communities are reported in Fig. 3-5. The genres that are shared by at least 50% of the nodes in each community are reported. As it can be seen from the figures, the proposed algorithm is capable of revealing communities that share similar genres. For example, in Fig. 3, 100% of the directors in $\mathcal{C}_1^1$ and $\mathcal{C}_2^1$ directed comedy and drama genres, respectively. Similarly, in Fig. 4, 100% of the movies in $\mathcal{C}_1^2$, $\mathcal{C}_2^2$, $\mathcal{C}_3^2$ and $\mathcal{C}_4^2$ are in the comedy, crime, drama and war/drama, respectively. With respect to the inter-layer communities, it was observed that in one of the communities, $\mathcal{C}_1^{12}$, the majority of the movies and directors shared the same genre with 94.74% of the movies and directors sharing the family genre. In the second inter-layer community, 60.7% of the nodes corresponded to movies directed by the directors in that community.

VI. CONCLUSION

In this paper, a new joint nonnegative matrix factorization approach is introduced to detect the community structure of multi-layer networks. The proposed approach models the multi-layer network as the union of a multiplex network and a bipartite network. The multiplex network encodes the intra-layer interactions while the bipartite network encodes the inter-layer interactions. The proposed algorithm aims to detect intra- and inter-layer communities, simultaneously.

The performance of the proposed approach is validated on both simulated and real-world multi-layer networks and compared to other existing algorithms. The simulation results show that the proposed algorithm outperforms existing algorithms in detecting the community structure of the network. In particular, the proposed algorithm can detect the community structure in both homogeneous and heterogeneous binary and weighted multi-layer networks by optimizing the same objective function. Moreover, the proposed algorithm is robust to noise and outliers and can reveal the community structure of the underlying network even when it consists of small communities. Furthermore, the proposed method combines computational efficiency and accuracy in detecting the community structure of the multi-layer network. Extensive application of the proposed algorithm to real networks from multiple disciplines shows that it performs well for both homogeneous and heterogeneous multi-layer networks and the detected communities are in agreement with the available metadata. Finally, future work will consider extending the proposed approach to M-layer networks.

APPENDIX A
THE $\mathbf{U}_\alpha$-SUBPROBLEM

The $\mathbf{U}_\alpha$-subproblem in Eq. 17 can be rewritten using the definition of the Frobenious norm and $L_{2,1}$-norm as follows:

$$\begin{aligned}\min_{\mathbf{U}_\alpha}\ \mathcal{J}_1 = \min_{\mathbf{U}_\alpha}\ & Tr(\mathbf{A}^\alpha\mathbf{Z}^\alpha\mathbf{A}^{\alpha\top} \\ & - \mathbf{A}^\alpha\mathbf{Z}^\alpha\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{Z}^\alpha\mathbf{A}^\alpha + \mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{Z}^\alpha\mathbf{U}_\alpha\mathbf{U}_\alpha^\top) \\ & + \mu_2 Tr(\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{U}_\alpha\mathbf{U}_\alpha^\top - 2\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{B}_\alpha\mathbf{B}_\alpha^\top), \\ s.t\ \mathbf{U}_\alpha \geq 0,\end{aligned} \tag{35}$$

where Tr is the trace function and $\mathbf{Z}_{ii}^\alpha \in \mathbb{R}^{n^\alpha \times n^\alpha} = 1/\|\mathbf{a}_i^\alpha - \mathbf{U}_\alpha\mathbf{u}_{\alpha i}^\top\|$. The optimization subproblem in Eq. 35 can be solved by following the standard procedure used in NMF to find the multiplicative updating rules. The gradient of the objective function, $\mathcal{J}_1$, with respect to $\mathbf{U}_\alpha$ can be calculated as follows:

$$\begin{aligned}\frac{\partial\mathcal{J}_1}{\partial\mathbf{U}_\alpha} = & -2\mathbf{Z}^\alpha\mathbf{A}^\alpha\mathbf{U}_\alpha - 2\mathbf{A}^\alpha\mathbf{Z}^\alpha\mathbf{U}_\alpha + 2\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{Z}^\alpha\mathbf{U}_\alpha \\ & + 2\mathbf{Z}^\alpha\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{U}_\alpha + 2\mu_2'\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{U}_\alpha - 2\mu_2'\mathbf{B}_\alpha\mathbf{B}_\alpha^\top\mathbf{U}_\alpha.\end{aligned} \tag{36}$$

Using the Karush-Kuhn Tucker (KKT) complementarity conditions, a static point can be reached where the KKT condition for the factor $\mathbf{U}_\alpha$ can be defined as:

$$\frac{\partial\mathcal{J}_1}{\partial(\mathbf{U}_\alpha)_{ij}}(\mathbf{U}_\alpha)_{ij} = 0, \tag{37}$$

which leads to the update rule of the $\mathbf{U}_\alpha$ factor as:

$$(\mathbf{U}_\alpha)_{ij} \leftarrow \frac{(\mathbf{Z}^\alpha\mathbf{A}^\alpha\mathbf{U}_\alpha+\mathbf{A}^\alpha\mathbf{Z}^\alpha\mathbf{U}_\alpha+\mu_2'\mathbf{B}_\alpha\mathbf{B}_\alpha^\top\mathbf{U}_\alpha)_{ij}(\mathbf{U}_\alpha)_{ij}}{(\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{Z}^\alpha\mathbf{U}_\alpha+\mathbf{Z}^\alpha\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{U}_\alpha+\mu_2'\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{U}_\alpha)_{ij}}, \quad (38)$$

where the update rule can be expressed in matrix form as:

$$\mathbf{U}_\alpha \leftarrow \mathbf{U}_\alpha \odot \Big((\mathbf{Z}^\alpha\mathbf{A}^\alpha\mathbf{U}_\alpha + \mathbf{A}^\alpha\mathbf{Z}^\alpha\mathbf{U}_\alpha + \mu_2'\mathbf{B}_\alpha\mathbf{B}_\alpha^\top\mathbf{U}_\alpha)) \oslash (\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{Z}^\alpha\mathbf{U}_\alpha + \mathbf{Z}^\alpha\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{U}_\alpha + \mu_2'\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{U}_\alpha)\Big), \quad (39)$$

where $\odot$ and $\oslash$ denote the Hadamard product and division, respectively.

APPENDIX B
THE $\mathbf{B}_\alpha$-SUBPROBLEM

Starting with the $\mathbf{B}_1$-subproblem in Eq. 18 which can be rewritten as follows:

$$\min_{\mathbf{B}_1} \mathcal{J}_2 = \min_{\mathbf{B}_1} \mu_1 Tr(\mathbf{A}^{12}\mathbf{Z}^{12}\mathbf{A}^{12^\top} - 2\mathbf{B}_1\mathbf{S}^{12}\mathbf{B}_2^\top\mathbf{Z}^{12}\mathbf{A}^{12^\top} + \mathbf{B}_1\mathbf{S}^{12}\mathbf{B}_2^\top\mathbf{Z}^{12}\mathbf{B}_2\mathbf{S}^{12^\top}\mathbf{B}_1^\top + \mu_2 Tr(-2\mathbf{U}_1\mathbf{U}_1^\top\mathbf{B}_1\mathbf{B}_1^\top + \mathbf{B}_1\mathbf{B}_1^\top\mathbf{B}_1\mathbf{B}_1^\top), \quad s.t\ \mathbf{B}_1 \geq 0, \quad (40)$$

where $\mathbf{Z}_{ii}^{12} \in \mathbb{R}^{n^2\times n^2} = 1/\|\mathbf{a}_i^{12} - \mathbf{B}_1\mathbf{S}^{12}\mathbf{b}_{2i}^\top\|$. Following the same procedure of the $\mathbf{U}_\alpha$-subproblem, the gradient of the objective function, $\mathcal{J}_2$, with respect to $\mathbf{B}_1$ can be found as:

$$\frac{\partial\mathcal{J}_2}{\partial\mathbf{B}_1} = -2\mu_1\mathbf{A}^{12}\mathbf{Z}^{12}\mathbf{B}_2\mathbf{S}^{12^\top} + 2\mu_1\mathbf{B}_1\mathbf{S}^{12}\mathbf{B}_2^\top\mathbf{Z}^{12}\mathbf{B}_2\mathbf{S}^{12^\top} - 4\mu_2\mathbf{U}_1\mathbf{U}_1^\top\mathbf{B}_1 + 4\mu_2\mathbf{B}_1\mathbf{B}_1^\top\mathbf{B}_1. \quad (41)$$

Similar to $\mathbf{U}_\alpha$, we use the KKT conditions to derive the update rule for $\mathbf{B}_1$,

$$\frac{\partial\mathcal{J}_2}{\partial(\mathbf{B}_1)_{ij}}(\mathbf{B}_1)_{ij} = 0, \quad (42)$$

then,

$$(\mathbf{B}_1)_{ij} \leftarrow \frac{(\mu_1\mathbf{A}^{12}\mathbf{Z}^{12}\mathbf{B}_2\mathbf{S}^{12^\top} + \mu_2'\mathbf{U}_1\mathbf{U}_1^\top\mathbf{B}_1)_{ij}(\mathbf{B}_1)_{ij}}{(\mu_1\mathbf{B}_1\mathbf{S}^{12}\mathbf{B}_2^\top\mathbf{Z}^{12}\mathbf{B}_2\mathbf{S}^{12^\top} + \mu_2'\mathbf{B}_1\mathbf{B}_1^\top\mathbf{B}_1)_{ij}}, \quad (43)$$

where $\mu_2' = 2\mu_2$. Similarly, the update rule of $\mathbf{B}_2$ can be found as:

$$(\mathbf{B}_2)_{ij} \leftarrow \frac{(\mu_1\mathbf{Z}^{12}\mathbf{A}^{12^\top}\mathbf{B}_1\mathbf{S}^{12} + \mu_2'\mathbf{U}_2\mathbf{U}_2^\top\mathbf{B}_2)_{ij}(\mathbf{B}_1)_{ij}}{(\mu_1\mathbf{Z}^{12}\mathbf{B}_2\mathbf{S}^{12^\top}\mathbf{B}_1^\top\mathbf{B}_1\mathbf{S}^{12} + \mu_2'\mathbf{B}_2\mathbf{B}_2^\top\mathbf{B}_2)_{ij}}. \quad (44)$$

Both update rules in Eq.43 and Eq. 44 can expressed in matrix form, respectively, as:

$$\mathbf{B}_1 \leftarrow \mathbf{B}_1 \odot \Big((\mu_1\mathbf{A}^{12}\mathbf{Z}^{12}\mathbf{B}_2\mathbf{S}^{12^\top} + \mu_2'\mathbf{U}_1\mathbf{U}_1^\top\mathbf{B}_1) \oslash (\mu_1\mathbf{B}_1\mathbf{S}^{12}\mathbf{B}_2^\top\mathbf{Z}^{12}\mathbf{B}_2\mathbf{S}^{12^\top} + \mu_2'\mathbf{B}_1\mathbf{B}_1^\top\mathbf{B}_1)\Big), \quad (45)$$

and

$$\mathbf{B}_2 \leftarrow \mathbf{B}_2 \odot \Big((\mu_1\mathbf{Z}^{12}\mathbf{A}^{12^\top}\mathbf{B}_1\mathbf{S}^{12} + \mu_2'\mathbf{U}_2\mathbf{U}_2^\top\mathbf{B}_2) \oslash (\mu_1\mathbf{Z}^{12}\mathbf{B}_2\mathbf{S}^{12^\top}\mathbf{B}_1^\top\mathbf{B}_1\mathbf{S}^{12} + \mu_2'\mathbf{B}_2\mathbf{B}_2^\top\mathbf{B}_2)\Big), \quad (46)$$

APPENDIX C
THE $\mathbf{S}^{12}$-SUBPROBLEM

The $\mathbf{S}^{12}$-subproblem in Eq. 20 is rewritten as:

$$\min_{\mathbf{S}^{12}} \mathcal{J}_4 = \min_{\mathbf{S}^{12}} \mu_1\ Tr(-2\mathbf{B}_1\mathbf{S}^{12}\mathbf{B}_2^\top\mathbf{Z}^{12}\mathbf{A}^{12^\top} + \mathbf{B}_1\mathbf{S}^{12}\mathbf{B}_2^\top\mathbf{M}^{12}\mathbf{B}_2\mathbf{S}^{12^\top}\mathbf{B}_1^\top), \quad s.t\ \mathbf{S}^{12} \geq 0. \quad (47)$$

The last update rule can be computed similar to the previous factors as follows,

$$\frac{\partial\mathcal{J}_4}{\partial\mathbf{S}_{ij}^{12}} = -2\mu_1\mathbf{B}_1^\top\mathbf{A}^{12}\mathbf{B}_2 + 2\mu_1\mathbf{B}_1^\top\mathbf{B}_1\mathbf{S}^{12}\mathbf{B}_2^\top\mathbf{B}_2), \quad (48)$$

where the KKT condition for $\mathbf{S}^{12}$ is defined as:

$$\frac{\partial\mathcal{J}_4}{\partial(\mathbf{S}_{ij}^{12})}(\mathbf{S}^{12})_{ij} = 0, \quad (49)$$

and results in the following update rule:

$$(\mathbf{S}^{12})_{ij} \leftarrow \frac{(\mathbf{B}_1^\top\mathbf{A}^{12}\mathbf{Z}^{12}\mathbf{B}_2)_{ij}}{(\mathbf{B}_1^\top\mathbf{B}_1\mathbf{S}^{12}\mathbf{B}_2^\top\mathbf{Z}^{12}\mathbf{B}_2)_{ij}}(\mathbf{S}^{12})_{ij}, \quad (50)$$

or in matrix form as:

$$\mathbf{S}^{12} \leftarrow \mathbf{S}^{12} \odot \Big((\mathbf{B}_1^\top\mathbf{A}^{12}\mathbf{Z}^{12}\mathbf{B}_2) \oslash (\mathbf{B}_1^\top\mathbf{B}_1\mathbf{S}^{12}\mathbf{B}_2^\top\mathbf{Z}^{12}\mathbf{B}_2)\Big) \quad (51)$$

APPENDIX D
THE PROOF THAT THE FUNCTION $\mathcal{L}(u_\alpha, (u_\alpha)_{ij}^t)$ IS AN AUXILIARY FUNCTION OF $\mathcal{J}(u_\alpha)$

Proof: For $\mathcal{L}(u_\alpha, (u_\alpha)_{ij}^t)$ to be an auxiliary function of $\mathcal{J}(u_\alpha)$, it should satisfy both conditions given in **Definition** 3. By comparing the Taylor series expansion of $\mathcal{J}(u_\alpha)$ in Eq. 30 with $\mathcal{L}(u_\alpha, (u_\alpha)_{ij}^t)$ defined in Eq. 31, it is obvious that the first two terms in $\mathcal{L}(u_\alpha, (u_\alpha)_{ij}^t)$ are greater than the first two terms in $\mathcal{J}(u_\alpha)$. Now, we have to prove the following:

$$\Big[6\frac{\big(\mathbf{U}_\alpha^t\mathbf{U}_\alpha^{t\top}\mathbf{Z}^\alpha\mathbf{U}_\alpha^t + \mathbf{Z}^\alpha\mathbf{U}_\alpha^t\mathbf{U}_\alpha^{t\top}\mathbf{U}_\alpha^t\big)_{ij}}{(u_\alpha)_{ij}^t} + 12\frac{\big(\mu_2\mathbf{U}_\alpha^t\mathbf{U}_\alpha^{t\top}\mathbf{U}_\alpha^t\big)_{ij}}{(u_\alpha)_{ij}^t}\Big] \geq \mathcal{J}''(u_\alpha). \quad (52)$$

First, we need to prove the following:

$$6\frac{\big(\mathbf{U}_\alpha^t\mathbf{U}_\alpha^{t\top}\mathbf{Z}^\alpha\mathbf{U}_\alpha^t\big)_{ij}}{\big(u_\alpha\big)_{ij}^t} \geq 2\Big[\big(\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{Z}^\alpha\big)_{ii} + \big(\mathbf{U}_\alpha^\top\mathbf{Z}^\alpha\big)_{ij}\big(u_\alpha\big)_{ij} + z_{ii}^\alpha\sum_p\big(u_\alpha\big)_{pj}^2\Big]. \quad (53)$$

Since $\mathbf{U}_\alpha$ and $\mathbf{Z}^\alpha$ are nonnegative, we have:

$$\frac{\big(\mathbf{U}_\alpha^t\mathbf{U}_\alpha^{t\top}\mathbf{Z}^\alpha\mathbf{U}_\alpha^t\big)_{ij}}{\big(u_\alpha\big)_{ij}^t} = \frac{\sum_p\big(\mathbf{U}_\alpha^t\mathbf{U}_\alpha^{t\top}\mathbf{Z}^\alpha\big)_{ip}\big(u_\alpha\big)_{pj}}{\big(u_\alpha\big)_{ij}^t} \geq \big(\mathbf{U}_\alpha\mathbf{U}_\alpha^\top\mathbf{Z}^\alpha\big)_{ii}. \quad (54)$$

$$\frac{\left(\mathbf{U}_\alpha^t \mathbf{U}_\alpha^{t\top} \mathbf{Z}^\alpha \mathbf{U}_\alpha^t\right)_{ij}}{\left(u_\alpha\right)_{ij}^t} = \frac{\sum_p \sum_q \left(\mathbf{U}_\alpha^t \mathbf{U}_\alpha^{t\top}\right)_{iq} \left(z^\alpha\right)_{qp} \left(u_\alpha\right)_{pj}}{\left(u_\alpha\right)_{ij}^t} \geq \left(\mathbf{U}_\alpha^\top \mathbf{Z}^\alpha\right)_{ij} \left(u_\alpha\right)_{ij}. \quad (55)$$

$$\frac{\left(\mathbf{U}_\alpha^t \mathbf{U}_\alpha^{t\top} \mathbf{Z}^\alpha \mathbf{U}_\alpha^t\right)_{ij}}{\left(u_\alpha\right)_{ij}^t} = \frac{\sum_{p,q,l} \left(u_\alpha^t\right)_{il} \left(u_\alpha^t\right)_{ql} \left(z^\alpha\right)_{qp} \left(u_\alpha\right)_{pj}}{\left(u_\alpha\right)_{ij}^t} \geq z_{ii}^\alpha \sum_p \left(u_\alpha\right)_{pj}^2. \quad (56)$$

Since Eq. 54 - Eq. 56 hold, Eq. 53 is satisfied. Second, we need to prove the following inequality:

$$6\frac{\left(\mathbf{Z}^\alpha \mathbf{U}_\alpha^t \mathbf{U}_\alpha^{t\top} \mathbf{U}_\alpha^t\right)_{ij}}{\left(u_\alpha\right)_{ij}^t} \geq 2\Bigg[\left(\mathbf{Z}^\alpha \mathbf{U}_\alpha \mathbf{U}_\alpha^\top\right)_{ii} + \left(\mathbf{Z}^\alpha \mathbf{U}_\alpha\right)_{ij} \left(u_\alpha\right)_{ij} + z_{ii}^\alpha \sum_p \left(u_\alpha\right)_{pj}^2\Bigg]. \quad (57)$$

Since $\mathbf{U}_\alpha$ and $\mathbf{Z}^\alpha$ are nonnegative, then:

$$\frac{\left(\mathbf{Z}^\alpha \mathbf{U}_\alpha^t \mathbf{U}_\alpha^{t\top} \mathbf{U}_\alpha^t\right)_{ij}}{\left(u_\alpha\right)_{ij}^t} = \frac{\sum_p \left(\mathbf{Z}^\alpha \mathbf{U}_\alpha^t \mathbf{U}_\alpha^{t\top}\right)_{ip} \left(u_\alpha^t\right)_{pj}}{\left(u_\alpha\right)_{ij}^t} \geq \left(\mathbf{Z}^\alpha \mathbf{U}_\alpha \mathbf{U}_\alpha^\top\right)_{ii}. \quad (58)$$

$$\frac{\left(\mathbf{Z}^\alpha \mathbf{U}_\alpha^t \mathbf{U}_\alpha^{t\top} \mathbf{U}_\alpha^t\right)_{ij}}{\left(u_\alpha\right)_{ij}^t} = \frac{\sum_p \sum_q \left(\mathbf{Z}^\alpha \mathbf{U}_\alpha^t\right)_{iq} \left(u_\alpha^t\right)_{pq} \left(u_\alpha^t\right)_{pj}}{\left(u_\alpha\right)_{ij}^t} \geq \left(\mathbf{Z}^\alpha \mathbf{U}_\alpha\right)_{ij} \left(u_\alpha\right)_{ij}. \quad (59)$$

$$\frac{\left(\mathbf{Z}^\alpha \mathbf{U}_\alpha^t \mathbf{U}_\alpha^{t\top} \mathbf{U}_\alpha^t\right)_{ij}}{\left(u_\alpha\right)_{ij}^t} = \frac{\sum_{p,q,l} \left(z^\alpha\right)_{il} \left(u_\alpha^t\right)_{lq} \left(u_\alpha^t\right)_{pq} \left(u_\alpha^t\right)_{pj}}{\left(u_\alpha\right)_{ij}^t} \geq z_{ii}^\alpha \sum_p \left(u_\alpha\right)_{ij}^2. \quad (60)$$

As a results, the inequality in Eq. 57 holds. Finally, we need to prove the following:

$$12\frac{\left(\mu_2 \mathbf{U}_\alpha^t \mathbf{U}_\alpha^{t\top} \mathbf{U}_\alpha^t\right)_{ij}}{\left(u_\alpha\right)_{ij}^t} \geq 4\mu_2\Bigg[\left(\mathbf{U}_\alpha \mathbf{U}_\alpha^\top\right)_{ij} + \left(\mathbf{U}_\alpha\right)_{ij} \left(u_\alpha\right)_{ij} + \sum_p \left(u_\alpha\right)_{pj}^2\Bigg]. \quad (61)$$

Following similar steps to the previous two proofs, we have:

$$\frac{\left(\mathbf{U}_\alpha^t \mathbf{U}_\alpha^{t\top} \mathbf{U}_\alpha^t\right)_{ij}}{\left(u_\alpha\right)_{ij}^t} = \frac{\sum_p \left(\mathbf{U}_\alpha^t \mathbf{U}_\alpha^{t\top}\right)_{ip} \left(u_\alpha^t\right)_{pj}}{\left(u_\alpha\right)_{ij}^t} \geq \left(\mathbf{U}_\alpha \mathbf{U}_\alpha^\top\right)_{ij}. \quad (62)$$

$$\frac{\left(\mathbf{U}_\alpha^t \mathbf{U}_\alpha^{t\top} \mathbf{U}_\alpha^t\right)_{ij}}{\left(u_\alpha\right)_{ij}^t} = \frac{\sum_p \sum_q \left(\mathbf{U}_\alpha^t\right)_{iq} \left(u_\alpha^t\right)_{pq} \left(u_\alpha^t\right)_{pj}}{\left(u_\alpha\right)_{ij}^t} \geq \left(\mathbf{U}_\alpha\right)_{ij} \left(u_\alpha\right)_{ij}. \quad (63)$$

$$\frac{\left(\mathbf{U}_\alpha^t \mathbf{U}_\alpha^{t\top} \mathbf{U}_\alpha^t\right)_{ij}}{\left(u_\alpha\right)_{ij}^t} = \frac{\sum_p \sum_q \left(u_\alpha^t\right)_{iq} \left(u_\alpha^t\right)_{pq} \left(u_\alpha^t\right)_{pj}}{\left(u_\alpha\right)_{ij}^t} \geq \sum_p \left(u_\alpha\right)_{pj}^2, \quad (64)$$

and this proves Eq. 61. Consequently, Eq. 52 holds and $\mathcal{L}(u_\alpha, (u_\alpha)_{ij}^t)$ is an auxiliary function of $\mathcal{J}(u_\alpha)$. □

ACKNOWLEDGMENT

The authors would like to thank to their colleague, Abdullah Karaaslanli, for sharing the multilayer Girvan Neman benchmark generation code.

REFERENCES

[1] A.-L. Barabási, "Network science," *Phil. Trans. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 371, no. 1987, 2013, Art. no. 20120375.
[2] E. Al-Sharoa, M. A. Al-Khassaweneh, and S. Aviyente, "Detecting and tracking community structure in temporal networks: A low-rank + sparse estimation based evolutionary clustering approach," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 5, no. 4, pp. 723–738, Dec. 2019.
[3] M. De Domenico, M. A. Porter, and A. Arenas, "MuxViz: A tool for multilayer analysis and visualization of networks," *J. Complex Netw.*, vol. 3, no. 2, pp. 159–176, 2015.
[4] M. De Domenico, C. Granell, M. A. Porter, and A. Arenas, "The physics of spreading processes in multilayer networks," *Nature Phys.*, vol. 12, no. 10, pp. 901–906, 2016.
[5] M. Kivelä, A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno, and M. A. Porter, "Multilayer networks," *J. Complex Netw.*, vol. 2, no. 3, pp. 203–271, 2014.
[6] X. Huang, D. Chen, T. Ren, and D. Wang, "A survey of community detection methods in multilayer networks," *Data Mining Knowl. Discovery*, vol. 35, no. 1, pp. 1–45, Jan. 2021.
[7] 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.
[8] Z. Kuncheva and G. Montana, "Community detection in multiplex networks using locally adaptive random walks," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining*, Aug. 2015, pp. 1308–1315.
[9] M. De Domenico, A. Lancichinetti, A. Arenas, and M. Rosvall, "Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems," *Phys. Rev. X*, vol. 5, no. 1, Mar. 2015, Art. no. 011027.
[10] A. Amelio and C. Pizzuti, "Community detection in multidimensional networks," in *Proc. IEEE 26th Int. Conf. Tools Artif. Intell.*, Nov. 2014, pp. 352–359.
[11] P. J. Mucha, T. Richardson, K. Macon, M. A. Porter, and J.-P. Onnela, "Community structure in time-dependent, multiscale, and multiplex networks," *Science*, vol. 328, no. 5980, pp. 876–878, May 2010.
[12] G. Zhu and K. Li, "A unified model for community detection of multiplex networks," in *Proc. Int. Conf. Web Inf. Syst. Eng.* Thessaloniki, Greece: Springer, 2014, pp. 31–46.
[13] E. Al-sharoa, M. Al-Khassaweneh, and S. Aviyente, "Tensor based temporal and multilayer community detection for studying brain dynamics during resting state fMRI," *IEEE Trans. Biomed. Eng.*, vol. 66, no. 3, pp. 695–709, Mar. 2019.
[14] S. Fortunato and D. Hric, "Community detection in networks: A user guide," *Phys. Rep.*, vol. 659, pp. 1–44, Nov. 2016.
[15] D. Jin, Z. Yu, P. Jiao, S. Pan, D. He, J. Wu, P. Yu, and W. Zhang, "A survey of community detection approaches: From statistical modeling to deep learning," *IEEE Trans. Knowl. Data Eng.*, early access, Aug. 11, 2021, doi: 10.1109/TKDE.2021.3104155.
[16] 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.
[17] M. Magnani, O. Hanteer, R. Interdonato, L. Rossi, and A. Tagarelli, "Community detection in multiplex networks," *ACM Comput. Surv.*, vol. 54, no. 3, pp. 1–35, Jun. 2021.
[18] L. Tang, X. Wang, and H. Liu, "Uncoverning groups via heterogeneous interaction analysis," in *Proc. 9th IEEE Int. Conf. Data Mining*, Dec. 2009, pp. 503–512.
[19] M. Berlingerio, M. Coscia, and F. Giannotti, "Finding and characterizing communities in multidimensional networks," in *Proc. Int. Conf. Adv. Social Netw. Anal. Mining*, Jul. 2011, pp. 490–494.
[20] P.-Y. Chen and A. O. Hero, "Multilayer spectral graph clustering via convex layer aggregation: Theory and algorithms," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 3, no. 3, pp. 553–567, Sep. 2017.

[21] D. Taylor, R. S. Caceres, and P. J. Mucha, "Super-resolution community detection for layer-aggregated multilayer networks," *Phys. Rev. X*, vol. 7, no. 3, Sep. 2017, Art. no. 031056.
[22] S. Venturini, A. Cristofari, F. Rinaldi, and F. Tudisco, "Louvain-like methods for community detection in multiplex networks," 2021, *arXiv:2106.13543*.
[23] A. Amelio, G. Mangioni, and A. Tagarelli, "Modularity in multilayer networks using redundancy-based resolution and projection-based inter-layer coupling," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 3, pp. 1198–1214, Jul. 2020.
[24] F. Karimi, S. Lotfi, and H. Izadkhah, "Multiplex community detection in complex networks using an evolutionary approach," *Expert Syst. Appl.*, vol. 146, May 2020, Art. no. 113184.
[25] H. Zhang, C.-D. Wang, J.-H. Lai, and S. Y. Philip, "Modularity in complex multilayer networks with multiple aspects: A static perspective," *Appl. Informat.*, vol. 4, pp. 1–29, May 2017.
[26] S. Pramanik, R. Tackx, A. Navelkar, J.-L. Guillaume, and B. Mitra, "Discovering community structure in multilayer networks," in *Proc. IEEE Int. Conf. Data Sci. Adv. Anal. (DSAA)*, Oct. 2017, pp. 611–620.
[27] S. Fortunato and M. Barthélemy, "Resolution limit in community detection," *Proc. Nat. Acad. Sci. USA*, vol. 104, no. 1, pp. 36–41, 2007.
[28] Z. Roozbahani, H. Emamgholizadeh, J. Rezaeenour, and M. Hajialikhani, "A systematic survey on multi-relational community detection," 2021, *arXiv:2103.15698*.
[29] X. Dong, P. Frossard, P. Vandergheynst, and N. Nefedov, "Clustering with multi-layer graphs: A spectral perspective," *IEEE Trans. Signal Process.*, vol. 60, no. 11, pp. 5820–5831, Nov. 2012.
[30] C. Chen, M. Ng, and S. Zhang, "Block spectral clustering for multiple graphs with inter-relation," *Netw. Model. Anal. Health Informat. Bioinf.*, vol. 6, no. 1, p. 8, Dec. 2017.
[31] B. Wang, A. M. Mezlini, F. Demir, M. Fiume, Z. Tu, M. Brudno, B. Haibe-Kains, and A. Goldenberg, "Similarity network fusion for aggregating data types on a genomic scale," *Nature Methods*, vol. 11, no. 3, pp. 333–337, Jan. 2014.
[32] Y. Wang and X. Qian, "Biological network clustering by robust NMF," in *Proc. 5th ACM Conf. Bioinf., Comput. Biol., Health Informat.*, Sep. 2014, pp. 688–689.
[33] W. Cheng, X. Zhang, Z. Guo, Y. Wu, P. F. Sullivan, and W. Wang, "Flexible and robust co-regularized multi-domain graph clustering," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2013, pp. 320–328.
[34] W. Tang, Z. Lu, and I. S. Dhillon, "Clustering with multiple graphs," in *Proc. 9th IEEE Int. Conf. Data Mining*, Dec. 2009, pp. 1016–1021.
[35] 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*, 2013, pp. 899–905.
[36] L. Gauvin, A. Panisson, and C. Cattuto, "Detecting the community structure and activity patterns of temporal networks: A non-negative tensor factorization approach," *PLoS ONE*, vol. 9, no. 1, Jan. 2014, Art. no. e86028.
[37] V. Gligorijević, 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.
[38] S. Javed, A. Mahmood, N. Werghi, K. Benes, and N. Rajpoot, "Multiplex cellular communities in multi-gigapixel colorectal cancer histology images for tissue phenotyping," *IEEE Trans. Image Process.*, vol. 29, pp. 9204–9219, 2020.
[39] 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.
[40] H. T. Nguyen, T. N. Dinh, and T. Vu, "Community detection in multiplex social networks," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Apr. 2015, pp. 654–659.
[41] S. Choi, "Algorithms for orthogonal nonnegative matrix factorization," in *Proc. IEEE Int. Joint Conf. Neural Netw., IEEE World Congr. Comput. Intell.*, Jun. 2008, pp. 1828–1832.
[42] Y.-X. Wang and Y.-J. Zhang, "Nonnegative matrix factorization: A comprehensive review," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 6, pp. 1336–1353, Jun. 2013.
[43] N. Gillis, *Nonnegative Matrix Factorization*. Philadelphia, PA, USA: SIAM, 2020.
[44] D. Kuang, S. Yun, and H. Park, "SymNMF: Nonnegative low-rank approximation of a similarity matrix for graph clustering," *J. Global Optim.*, vol. 62, no. 3, pp. 545–574, Jul. 2015.
[45] D. Kuang, C. Ding, and H. Park, "Symmetric nonnegative matrix factorization for graph clustering," in *Proc. SIAM Int. Conf. Data Mining*. Philadelphia, PA, USA: SIAM, Apr. 2012, pp. 106–117.
[46] B. Long, Z. Zhang, and P. S. Yu, "Co-clustering by block value decomposition," in *Proc. 11th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2005, pp. 635–640.
[47] D. Kong, C. Ding, and H. Huang, "Robust nonnegative matrix factorization using $L_{2,1}$-norm," in *Proc. 20th ACM Int. Conf. Inf. Knowl. Manage.*, 2011, pp. 673–682.
[48] U. von Luxburg, "A tutorial on spectral clustering," *Statist. Comput.*, vol. 17, no. 4, pp. 395–416, Dec. 2007.
[49] 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*. Philadelphia, PA, USA: SIAM, Apr. 2005, pp. 606–610.
[50] B. Long, X. Wu, Z. Zhang, and P. S. Yu, "Unsupervised learning on k-partite graphs," in *Proc. 12th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2006, pp. 317–326.
[51] J.-P. Brunet, P. Tamayo, T. R. Golub, and J. P. Mesirov, "Metagenes and molecular pattern discovery using matrix factorization," *Proc. Nat. Acad. Sci. USA*, vol. 101, no. 12, pp. 4164–4169, 2004.
[52] H. Kim and H. Park, "Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis," *Bioinformatics*, vol. 23, no. 12, pp. 1495–1502, May 2007.
[53] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *J. Stat. Mech., Theory Exp.*, vol. 2008, no. 10, Oct. 2008, Art. no. P10008.
[54] V. A. Traag, R. Aldecoa, and J.-C. Delvenne, "Detecting communities using asymptotical surprise," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 92, no. 2, Aug. 2015, Art. no. 022816.
[55] S. Kullback and R. Leibler, "On information and sufficiency," *Ann. Math. Statist.*, vol. 22, no. 1, pp. 79–86, 1951.
[56] M. L. Hartsperger, F. Blöchl, V. Stümpflen, and F. J. Theis, "Structuring heterogeneous biological information using fuzzy clustering of k-partite graphs," *BMC Bioinf.*, vol. 11, no. 1, pp. 1–15, Dec. 2010.
[57] C. Ding, T. Li, W. Peng, and H. Park, "Orthogonal nonnegative matrix t-factorizations for clustering," in *Proc. 12th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2006, pp. 126–135.
[58] 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.*, 2015, pp. 1–7.
[59] 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.
[60] A. Miyauchi and Y. Kawase, "What is a network community? A novel quality function and detection algorithms," in *Proc. 24th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2015, pp. 1471–1480.
[61] T. Chakraborty, A. Dalmia, A. Mukherjee, and N. Ganguly, "Metrics for community analysis: A survey," *ACM Compt. Surv.*, vol. 50, no. 4, pp. 1–37, 2017.
[62] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 13, 2001, pp. 556–562.
[63] A. Mahmood, M. Small, S. A. Al-Maadeed, and N. Rajpoot, "Using geodesic space density gradients for network community detection," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 4, pp. 921–935, Apr. 2017.
[64] A. Mahmood and M. Small, "Subspace based network community detection using sparse linear coding," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 3, pp. 801–812, Mar. 2016.
[65] 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, Jun. 2002.
[66] N. Eagle, A. S. Pentland, and D. Lazer, "Inferring friendship network structure by using mobile phone data," *Proc. Nat. Acad. Sci. USA*, vol. 106, no. 36, pp. 15274–15278, 2009.
[67] N. Kiukkonen, J. Blom, O. Dousse, D. Gatica-Perez, and J. Laurila, "Towards rich mobile phone datasets: Lausanne data collection campaign," in *Proc. ICPS*, vol. 68, no. 7, Berlin, Germany, 2010, pp. 1–7.
[68] E. Lazega, *The Collegial Phenomenon: The Social Mechanisms of Cooperation Among Peers in a Corporate Law Partnership*. London, U.K.: Oxford Univ. Press, 2001.
[69] T. A. B. Snijders, P. E. Pattison, G. L. Robins, and M. S. Handcock, "New specifications for exponential random graph models," *Sociol. Methodol.*, vol. 36, no. 1, pp. 99–153, 2006.

[70] L. G. S. Jeub, M. W. Mahoney, P. J. Mucha, and M. A. Porter, "A local perspective on community structure in multilayer networks," *Netw. Sci.*, vol. 5, no. 2, pp. 144–163, Jun. 2017.
[71] A. R. Pamfil, S. D. Howison, R. Lambiotte, and M. A. Porter, "Relating modularity maximization and stochastic block models in multilayer networks," *SIAM J. Math. Data Sci.*, vol. 1, no. 4, pp. 667–698, Jan. 2019.
[72] B. L. Chen, D. H. Hall, and D. B. Chklovskii, "Wiring optimization can relate neuronal structure and function," *Proc. Nat. Acad. Sci. USA*, vol. 103, no. 12, pp. 4723–4728, 2006.
[73] D. Dua and C. Graff, "UCI machine learning repository," School Inf. Comput. Sci., Univ. California, Irvine, CA, USA, 2019. [Online]. Available: http://archive.ics.uci.edu/ml
[74] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 594–611, Apr. 2006.
[75] Y. Li, F. Nie, H. Huang, and J. Huang, "Large-scale multi-view spectral clustering via bipartite graph," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015, pp. 1–7.

ESRAA M. AL-SHAROA (Member, IEEE) received the B.S. degree in electrical engineering from the Jordan University of Science and Technology, Irbid, Jordan, in 2005, and the M.S. and Ph.D. degrees in electrical engineering from Michigan State University, East Lansing, MI, USA, in 2007 and 2019, respectively. She is currently an Assistant Professor with the Department of Electrical Engineering, Jordan University of Science and Technology. Her research interests include signal processing, in particular, community detection and tracking in temporal and multilayer networks.

SELIN AVIYENTE (Senior Member, IEEE) received the B.S. degree in electrical and electronics engineering from Bogazici University, in 1997, and the M.S. and Ph.D. degrees in electrical engineering from the University of Michigan, in 1999 and 2002, respectively. She is currently a Professor with the Department of Electrical and Computer Engineering, Michigan State University. Her research interest includes the theory and applications of statistical signal processing, in particular non-stationary signal analysis. She is interested in developing methods for efficient signal representation, detection, and classification. She is also interested in the applications of signal processing to biological signals, such as the analysis of event related brain potentials. Her current research focuses on the study of the functional networks in the brain.