Multi-view Clustering with Graph Embedding for Connectome Analysis

Guixiang Ma*, Lifang He*, Chun-Ta Lu*, Weixiang Shao†,

Philip S. Yu¶,*, Alex D. Leow*, Ann B. Ragin‡

*University of Illinois at Chicago, Chicago, IL, USA

†Google Inc, Mountain View, USA

¶Shanghai Institute for Advanced Communication and Data Science, Fudan University, Shanghai, China

‡Northwestern University, Chicago, IL, USA

{gma4,clu29,psyu}@uic.edu,{lifanghescut,software.shao,alexfeuillet}@gmail.com,ann-ragin@northwestern.edu

ABSTRACT

Multi-view clustering has become a widely studied problem in the area of unsupervised learning. It aims to integrate multiple views by taking advantages of the consensus and complimentary information from multiple views. Most of the existing works in multi-view clustering utilize the vector-based representation for features in each view. However, in many real-world applications, instances are represented by graphs, where those vector-based models cannot fully capture the structure of the graphs from each view. To solve this problem, in this paper we propose a Multi-view Clustering framework on graph instances with Graph Embedding (MCGE). Specifically, we model the multi-view graph data as tensors and apply tensor factorization to learn the multi-view graph embeddings, thereby capturing the local structure of graphs. We build an iterative framework by incorporating multi-view graph embedding into the multi-view clustering task on graph instances, jointly performing multi-view clustering and multi-view graph embedding simultaneously. The multi-view clustering results are used for refining the multi-view graph embedding, and the updated multi-view graph embedding results further improve the multi-view clustering. Extensive experiments on two real brain network datasets (*i.e.*, HIV and Bipolar) demonstrate the superior performance of the proposed MCGE approach in multi-view connectome analysis for clinical investigation and application.

KEYWORDS

Multi-view Clustering, Graph Embedding, Connectome Analysis

1 INTRODUCTION

Advances in capabilities for data acquisition have given rise to an explosion of new information in the form of graph representations. These data are inherently represented as a set of nodes and links, instead of feature vectors as in traditional data. Brain networks, for example, are comprised of anatomic regions as nodes, and functional/structural connectivities between the brain regions as links. Linkage structures often come from different sources, called as multi-view data. For instance, fMRI (functional magnetic resonance imaging) and DTI (diffusion tensor imaging) are two major neuroimaging approaches widely used in neuroscience research and in clinical applications [8, 24, 47]. Connections in brain networks derived from fMRI brain images encode correlations in functional activity among brain regions, whereas DTI networks provide information concerning structural connections (*i.e.* white matter fiber paths) between different brain regions. The different networks afford two different views of the brain connectivity.

Multi-view clustering has received considerable attention for unlabeled data with multiple views from diverse domains. While there have been advances in multi-view clustering, most approaches are based on vector representation of features in each view and combining vectors from different views for the clustering task [21, 46]. However, the complex structures and the lack of vector representations within graph data, pose serious challenges for this kind of vector-based approach. It is desirable to find a way that can better capture and exploit graph structural information for multi-view clustering of graph instances. To address this problem, this paper explores an approach involving multi-view clustering of graph instances based on graph embedding and its application to connectome analysis in multi-view brain networks on HIV and Bipolar. The goal of graph embedding is to find low-dimensional representations of graphs that can preserve the inherent structure and properties [26, 45]. While graph embedding technology has been broadly used for graph mining, to the best of our knowledge, this approach has not been used for multi-view clustering of graph instances. There are two main challenges that must be addressed for the problem of multi-view clustering with graph embedding:

- How to learn the graph embedding for each graph instance with multiple views, such that the graph embeddings can encode the multi-view structure information of the graphs? Specifically, the embeddings of the similar nodes within the graph instance should be close.
- How to leverage the multi-view graph embedding results to facilitate the multi-view clustering task on graph instances? The graph embeddings mainly captures the local structure

CIKM'17 , November 6–10, 2017, Singapore, Singapore

ACM ISBN 978-1-4503-4918-5/17/11…$15.00

https://doi.org/10.1145/3132847.3132909

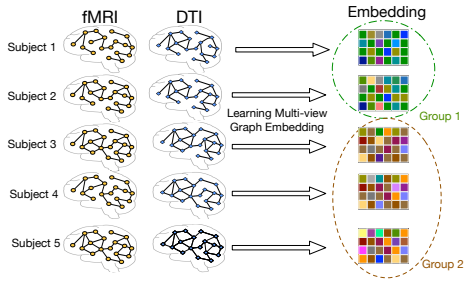


Figure 1: An example of the MCGE problem

of graphs, while the similarity between the graph instances holds their global structure information. For multi-view clustering on graph data, it is critical to appropriately fuse these two kinds of graph structure information .

To address the above challenges, we propose the MCGE (multi-view clustering with graph embedding) framework. Our contributions can be summarized as:

- We model the multi-view graph data as tensors, and apply tensor factorization to learn the multi-view embeddings of graphs. In this manner, the graph embeddings can capture the key local structure of the graphs in all the views, while also encoding the latent correlations between different views.
- We employ graph kernel to measure the similarity between graph instances in each view, construct a multi-view kernel tensor based on kernel matrices, and obtain the common latent factors that encode the global structure information.
- We propose to jointly perform the multi-view graph embedding stage and the multi-view clustering stage in an iterative manner. Considering the fact that the graphs clustered into the same group tend to have similar local structure, for each graph, we use the multi-view embeddings of the neighbour graphs clustered in the same group to refine its multi-view embedding. Then the updated multi-view embeddings of the graphs will be used for the multi-view clustering stage in the next iteration. Following this iterative two-stage process, the multi-view graph embedding and multi-view clustering will be improved until we obtain an optimal clustering results.
- We apply the proposed MCGE framework for unsupervised multi-view connectome analysis on HIV and Bipolar. Specifically, we study the connectome of fMRI and DTI brain networks and aim to cluster the subjects with similar neurological status into the same group as shown in Figure 1. Experimental results on the HIV and Bipolar datasets show the effectiveness of MCGE for multi-view clustering in connectome analysis.

The rest of this paper is organized as follows. In the next section, problem formulation and background are given. The details of the proposed MCGE framework are presented in Sections 3 and 4. Extensive experimental results and analysis are shown in Section 5. Related work is discussed in Section 6 and followed by the conclusion in Section 7.

Table 1: List of basic symbols.

Symbol	Definition and description
x	each lowercase letter represents a scale
$\mathbf{x}$	each boldface lowercase letter represents a vector
$\mathbf{X}$	each boldface uppercase letter represents a matrix
$\mathcal{X}$	each calligraphic letter represents a tensor
$\langle \cdot, \cdot \rangle$	denotes inner product
$\circ$	denotes tensor product (outer product)
$\otimes$	denotes Kronecker product
$\odot$	denotes Khatri-Rao product

2 PRELIMINARIES

In this section, we first introduce some notations and terminologies that we will use throughout the paper. Then we formulate the problem of interest formally.

Notations. Vectors are denoted by boldface lowercase letters, matrices are denoted by boldface capital letters, and tensors are denoted by calligraphic letters. An element of a vector $\mathbf{x}$, a matrix $\mathbf{X}$, or a tensor $\mathcal{X}$ is denoted by x_i, x_{ij}, x_{ijk}, etc., depending on the number of indices (also known as modes). For a matrix $\mathbf{X} \in \mathbb{R}^{n \times m}$, its i-th row and j-th column are denoted by $\mathbf{x}^i$ and $\mathbf{x}_j$, respectively. The Frobenius norm of $\mathbf{X}$ is defined as $\|\mathbf{X}\|_F = \sqrt{\sum_{i=1}^{n} \|\mathbf{x}^i\|_2^2}$. For any vector $\mathbf{x} \in \mathbb{R}^n$, $Diag(\mathbf{x}) \in \mathbb{R}^{n \times n}$ is the diagonal matrix whose diagonal elements are x_i. $\mathbf{I}_n$ denotes an identity matrix with size n. We denote an undirected graph as $G = (V, E)$, where V is the set of nodes and $E \subset V \times V$ is the set of edges. An overview of the basic symbols used in this paper can be found in Table 1.

Definition 2.1 (Tensor). An nth-order tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_n}$ is an element of the outer product of n vector spaces, each of which has its own coordinate system.

Definition 2.2 (Outer product). The outer product of vectors $\mathbf{x}^{(k)} \in \mathbb{R}^{I_k}$ for $k = 1, 2, \cdots, n$ is an n-th order tensor and defined elementwise by $\left(\mathbf{x}^{(1)} \circ \mathbf{x}^{(2)} \circ \cdots \circ \mathbf{x}^{(n)}\right)_{i_1, i_2, \cdots, i_n} = x_{i_1}^{(1)} x_{i_2}^{(2)} \cdots x_{i_n}^{(n)} = \prod_{k=1}^{n} x_{i_k}^{(k)}$ for all values of the indices.

Definition 2.3 (Kronecker Product). The Kronecker product of two matrices $\mathbf{A} \in \mathbb{R}^{I \times J}, \mathbf{B} \in \mathbb{R}^{K \times L}$ is a matrix in the dimension of $IK \times JL$:

$$\mathbf{A} \otimes \mathbf{B} = \begin{pmatrix} a_{11}\mathbf{B} & a_{12}\mathbf{B} & \cdots & a_{1J}\mathbf{B} \\ a_{21}\mathbf{B} & a_{22}\mathbf{B} & \cdots & a_{2J}\mathbf{B} \\ \vdots & \vdots & \vdots & \vdots \\ a_{I1}\mathbf{B} & a_{I2}\mathbf{B} & \cdots & a_{IJ}\mathbf{B} \end{pmatrix} \tag{1}$$

Definition 2.4 (Khatri-Rao Product). The Khatri-Rao product of two matrices $\mathbf{A} \in \mathbb{R}^{I \times K}, \mathbf{B} \in \mathbb{R}^{J \times K}$ is a matrix in dimension of $IJ \times K$:

$$\mathbf{A} \odot \mathbf{B} = (a_1 \otimes b_1, a_2 \otimes b_2, \cdots, a_K \otimes b_K) \tag{2}$$

where $a_1, a_2, \cdots, a_K$ are the columns of $\mathbf{A}$ and $b_1, b_2, \cdots, b_K$ are the columns of $\mathbf{B}$.

Definition 2.5 (Mode-k Matricization). The mode-k matricization of a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_n}$, denoted by $\mathbf{X}_{(k)} \in \mathbb{R}^{I_k \times J}$, where

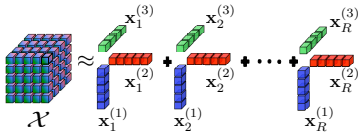


Figure 2: The CP Factorization for a third-order tensor $\mathcal{X}$

$J = \Pi_{q=1,q\neq k}^{n} I_q$. Each tensor element with indices $(i_1, i_2, \cdots, i_n)$ maps to a matrix element (i_k, j), such that

$$j = 1 + \sum_{p=1,p\neq k}^{m} (i_p - 1)J_p, \ with$$
$$J_p = \begin{cases} 1, & \text{if } p = 1 \text{ or } (p = 2 \text{ and } k = 1) \\ \Pi_{q=1,q\neq k}^{p-1} I_q, & \text{otherwise.} \end{cases} \tag{3}$$

Definition 2.6 (CP Factorization). For a general tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_n}$, its CANDECOMP / PARAFAC (CP) factorization is

$$\mathcal{X} = [\![\mathbf{X}^{(1)}, \cdots, \mathbf{X}^{(n)}]\!] \equiv \sum_{r=1}^{R} \mathbf{x}_r^{(1)} \circ \cdots \circ \mathbf{x}_r^{(n)}, \tag{4}$$

where for $k = 1, 2, \cdots, n$, $\mathbf{X}^{(k)} = [\mathbf{x}_1^{(k)}, \cdots, \mathbf{x}_R^{(k)}]$ are factor matrices of size $I_k \times R$, R is the number of factors, and $[\![\cdot]\!]$ is used for shorthand. Figure 2 shows the form of the CP Factorization for a third-order tensor example.

To obtain the CP factorization $[\![\mathbf{X}^{(1)}, \cdots, \mathbf{X}^{(n)}]\!]$, the objective is to minimize the following estimation error:

$$\mathcal{L} = \min_{\mathbf{X}^{(1)}, \cdots, \mathbf{X}^{(n)}} \|\mathcal{X} - [\![\mathbf{X}^{(1)}, \cdots, \mathbf{X}^{(n)}]\!]\|_F^2 \tag{5}$$

However, $\mathcal{L}$ is not jointly convex w.r.t. $\mathbf{X}^{(1)}, \cdots, \mathbf{X}^{(n)}$. A widely used optimization technique is the Alternating Least Squares (ALS) algorithm, which alternatively minimize $\mathcal{L}$ for each variable while fixing the other, that is,

$$\mathbf{X}^{(k)} \leftarrow \arg\min_{\mathbf{X}^{(k)}} \|\mathbf{X}_{(k)} - \mathbf{X}^{(k)} (\odot_{i\neq k}^{n} \mathbf{X}^{(i)})^{\mathrm{T}}\|_F^2 \tag{6}$$

where $\odot_{i\neq k}^{n} \mathbf{X}^{(i)} = \mathbf{X}^{(n)} \odot \cdots \mathbf{X}^{(k+1)} \odot \mathbf{X}^{(k-1)} \cdots \odot \mathbf{X}^{(1)}$.

Problem Definition We study the problem of multi-view clustering of graph instances with multi-view graph embedding. Assume we are given a set of instances $D = \{G_1, G_2, \cdots, G_n\}$ with v views, where each instance is represented with a graph with m nodes in each view. For the j-th view, we have a set of graphs with the affinity matrices $D^{(j)} = \{\mathbf{G}_1^{(j)}, \mathbf{G}_2^{(j)}, \cdots, \mathbf{G}_n^{(j)}\}$. The goal of multi-view clustering on D is to cluster the graphs in D into k subsets. Figure 1 shows a simple two-view example of the MCGE problem intuitively. Given the fMRI and DTI brain networks of five subjects, MCGE aims to learn multi-view graph embedding for each of them, and cluster these subjects into different groups based on the obtained multi-view graph embeddings.

3 MCGE FRAMEWORK

In this section, we first present the proposed MCGE framework consisting of two stages: multi-view graph embedding and multi-view clustering with graph embedding. Then we describe the optimization scheme of our framework.

3.1 Multi-view Graph Embedding

Graph embedding is an important tool in topological graph theory, which has been widely used in data analysis [2, 13, 45]. In the unsupervised situation, conventional methods for multi-view graph embedding either glued the graph affinity matrices from all the views together into a big graph [11, 14], or collaboratively explored the consensus embedding from different views (individual affinity matrices) [43, 44, 48]. However these methods can only capture the linear relationships in multi-view graph data. In order to achieve better embeddings, here we develop a multilinear embedding approach via tensorization as follows.

To model the multiple views for each graph instance G_i, we build a tensor $\mathcal{T}_i$ by stacking the graph affinity matrices from all the v views of the graph. Assume that the dimension of the row vectors in the graph embeddings is c, and let $\mathbf{F}_i \in \mathbb{R}^{m\times c}$ be the graph embedding of G_i, *i.e.*, the j-th row vector of $\mathbf{F}_i$ represent the embedding of node j on graph instance G_i. Then we can formulate the multi-view graph embedding as the following optimization problem based on CP factorization:

$$\min_{\mathbf{F_i},\mathbf{H_i}} \left\|\mathcal{T}_i - [\![\mathbf{F}_i, \mathbf{F}_i, \mathbf{H}_i]\!]\right\|_F^2$$
$$\text{s.t. } \mathbf{F}_i{}^{\mathrm{T}}\mathbf{F}_i = \mathbf{I}_c \tag{7}$$

where $\mathbf{F}_i \in \mathbf{R}^{m\times c}$ and $\mathbf{H}_i \in \mathbf{R}^{v\times c}$ are the latent factor matrices.

Besides, as we discussed earlier, the graphs clustered into the same group tend to have more similar local structure. That is, for two graphs in the same cluster, the closer they are, the more similar local structure they tend to have. Based on this assumption, we incorporate such global cluster information to further improve the multi-view graph embedding result in Equation (7). Assuming we can obtain a weight matrix $\mathbf{W}$, where w_{ij} denotes the weight of G_j for G_i and a larger w_{ij} implies a closer distance between G_i and G_j in the same cluster. By incorporating the weighted influence from the neighbor graphs into Equation (7), we have the following objective function:

$$\min_{\mathbf{F_i},\mathbf{H_i}} \left\|\mathcal{T}_i - [\![\mathbf{F}_i, \mathbf{F}_i, \mathbf{H}_i]\!]\right\|_F^2 + \beta\left\|\mathbf{F}_i - \sum_j w_{ij}\mathbf{F}_j\right\|_F^2$$
$$\text{s.t. } \mathbf{F}_i{}^{\mathrm{T}}\mathbf{F}_i = \mathbf{I}_c \tag{8}$$

where β is a parameter balancing the two parts.

In the following section, we will show how to incorporate the graph embeddings into the multi-view clustering framework and how to obtain the weight matrix $\mathbf{W}$ from the clustering results.

3.2 Multi-view Clustering via Graph Embedding

Since graph embedding usually encodes local structure of graphs, and the original affinity matrix holds the global structure, we propose to consider both of these two kinds of structure information for the multi-view clustering task. Specifically, we employ the graph kernel to measure the similarity of the global structure between different graphs. Graph kernel is a pervasive method for comparing graphs [37]. Here we employ the random walk graph kernel [37],

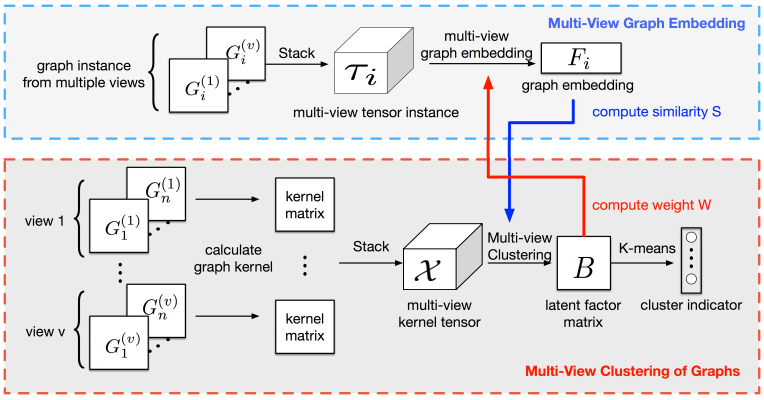


Figure 3: The framework of the proposed model MCGE.

which is one of the most widely used graph kernels, to measure the similarity between the affinity matrices of different graphs in each view. Since we have n graphs in v views, we will get v kernel matrices, each with dimension of $n \times n$. In order to integrate the multiple views, we propose to stack the v kernel matrices together, which form a tensor $\mathcal{X} \in \mathbf{R}^{n \times n \times v}$. Then we apply CP factorization on the tensor $\mathcal{X}$ to get the common factor matrices across all the views. Suppose the number of factors is k, $\mathcal{X}$ can be factorized as:

$$\mathcal{X} = [\![\mathbf{B}, \mathbf{B}, \mathbf{A}]\!] \tag{9}$$

where $\mathbf{B} \in \mathbf{R}^{n \times k}$ and $\mathbf{A} \in \mathbf{R}^{v \times k}$ are the latent factor matrices. Notably, $\mathbf{B}$ can be interpreted as the common latent factor across all the views, which can be used for clustering the graphs.

Now let us consider how to incorporate the results of multi-view graph embedding into the multi-view clustering stage. As we discussed above, the multi-view graph embeddings imply the local structure of the graphs, and graphs with similar local structure tend to be close to each other in the original multi-view feature space.

Suppose we have obtained a set of graph embeddings $F = \{\mathbf{F}_1, \mathbf{F}_2, \cdots, \mathbf{F}_n\}$, where $\mathbf{F}_i \in \mathbb{R}^{m \times c}$ is the multi-view graph embedding for G_i, we can build a similarity matrix $\mathbf{S} \in \mathbb{R}^{n \times n}$, where s_{ij} denotes the similarity between two examples G_i and G_j in terms of graph embedding, and we define it as:

$$s_{ij} = 1 - \left\|\mathbf{F}_i - \mathbf{F}_j\right\|_F^2 \tag{10}$$

Then we can formulate the following objective function on the basis of the spectral analysis [38]:

$$\begin{aligned} \min_{\mathbf{B}} \; &= \sum_{i,j=1}^{n} s_{ij} \left\| \frac{\mathbf{b}_i}{\sqrt{d_{ii}}} - \frac{\mathbf{b}_j}{\sqrt{d_{jj}}} \right\|_2^2 = \mathrm{Tr}\left(\mathbf{B}^{\mathrm{T}}\mathbf{L}\mathbf{B}\right) \\ &\text{s.t. } \mathbf{B}^{\mathrm{T}}\mathbf{B} = \mathbf{I}_k \end{aligned} \tag{11}$$

where $\mathbf{L} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{S})\mathbf{D}^{-\frac{1}{2}}$ is the symmetric normalized Laplacian matrix, and $\mathbf{D}$ is a diagonal matrix with $d_{ii} = \sum_{j=1}^{n} s_{ij}$.
By combining the above tensor CP factorization strategy with Equation (11), we can formulate the multi-view clustering task as the following optimization problem:

$$\begin{aligned} \min_{\mathbf{B},\mathbf{A}} \; &\left\|\mathcal{X} - [\![\mathbf{B}, \mathbf{B}, \mathbf{A}]\!]\right\|_F^2 + \alpha \mathrm{Tr}\left(\mathbf{B}^{\mathrm{T}}\mathbf{L}\mathbf{B}\right) \\ &\text{s.t. } \mathbf{B}^{\mathrm{T}}\mathbf{B} = \mathbf{I}_k \end{aligned} \tag{12}$$

where α is a parameter balancing two parts.

After we obtain matrix $\mathbf{B}$, we can apply k-means clustering on the row vectors of $\mathbf{B}$ and then we know which graphs are clustered into the same group and which ones are not. This result will help determine the weight matrix $\mathbf{W}$ for the multi-view graph embedding stage. Specifically, for graph G_i we consider the graphs from the same cluster with G_i, and we aim to infer the weights of influence they should have on G_i. Suppose we use $\mathbf{X}_i$ to represent both the global and local structure of G_i,then this problem can be formulated as the following minimization problem based on LLE method [31]:

$$\begin{aligned} \min_{\mathbf{W}} \; &\sum_i \left\|\mathbf{X}_i - \sum_j w_{ij}\mathbf{X}_j\right\|_F^2 \\ &\text{s.t.} \sum_j w_{ij} = 1 \end{aligned} \tag{13}$$

where w_{ij} denotes the weight of G_j for G_i, and $w_{ij} = 0$ if G_j and G_i are not in the same cluster. Note that there is no need for an explicit definition of $\mathbf{X}_i$ here, as it will be implicitly represented with both the affinity matrices and the multi-view graph embedding results, which will be used for the optimization of $\mathbf{W}$. The details will be illustrated in Section 4.

3.3 The Overall Framework: MCGE

With the two stages discussed above, we can formulate the overall iterative process for the MCGE framework. As the multi-view graph embedding and multi-view clustering depend on each other, we propose to jointly perform these two stages. In each iteration, we first perform the multi-view graph embedding on each graph, and then utilize the obtained graph embedding in the multi-view clustering stage. Then the resulted graph cluster information will be used for refining the multi-view graph embeddings in the next iteration. Following this alternate two-stage process, the multi-view graph embedding and multi-view clustering will be improved by each other until convergence.

An overview of our framework is shown in Figure 3. The upper part demonstrates the multi-view graph embedding stage in MCGE, and the lower part shows the multi-view clustering stage, while the blue arrow and red arrow indicate the interaction of the two stages. Overall, given a set of graph instances $D = \{G_1, G_2, \cdots, G_n\}$ with v views, we aim to obtain a multi-view graph embedding for each of these graph instances, and then use the multi-view graph embeddings as key features for clustering graph instances.

As shown in Figure 3, in the multi-view graph embedding stage, for each graph instance G_i, we stack its affinity matrices from all the v views together to form a multi-view tensor instance $\mathcal{T}_i$. Then we apply tensor factorization in Equation (8) to learn its multi-view embedding, which partially depend on the embeddings of the other graphs from the same cluster that is determined by the multi-view clustering stage. Meanwhile, in the multi-view clustering stage, we first measure the similarity between each pair of the graphs by calculating the graph kernel from each view, and then we stack the kernel matrices from all the views, resulting in a multi-view kernel tensor $\mathcal{X}$. By utilizing the CP factorization on $\mathcal{X}$, we can get the common factor $\mathbf{B}$ across all the views. Considering the importance of graph embedding in capturing graph structure, we compute the similarity between graphs based on the multi-view graph embedding results and incorporate it into the CP factorization scheme with a spectral analysis term, as shown in Equation (12). The latent factor $\mathbf{B}$ obtained from this step will indicate which graphs are closer to each other, thus can be further used for computing the weight matrix $\mathbf{W}$, which will be used for updating the multi-view graph embeddings in the next iteration. Vice versa, the new multi-view graph embeddings will be used for updating the similarity $\mathbf{S}$, thus improving the multi-view clustering stage.

4 OPTIMIZATION

Since the objective function in Equation (12) is not convex with respect to $\mathbf{A}$ and $\mathbf{B}$ jointly, and Equation (8) is not convex with respect to $\mathbf{F}_i$, there is no closed-form solution for such problem. We employ an Alternating Direction Method of Multipliers (ADMM) scheme [4, 40] to solve these problems, which alternately updates one variable while fixing others until convergence.

We first solve the optimization problem in Equation (12). The variables to be estimated include $\mathbf{B}$ and $\mathbf{A}$.

Update factor matrix B. We first update $\mathbf{B}$ while fixing $\mathbf{A}$. Due to the fourth-order term, the objective function in Equation (12) is not convex with respect to $\mathbf{B}$, thus being difficult to optimize. We employ the variable substitution technique to solve this problem. By substituting the second $\mathbf{B}$ with $\mathbf{P}$ in Equation (12), we obtain the equivalent form of Equation (12):

$$\begin{aligned} &\min_{\mathbf{B}} \|\mathcal{X} - [\![\mathbf{B}, \mathbf{P}, \mathbf{A}]\!]\|_F^2 + \alpha \mathrm{Tr}(\mathbf{B}^{\mathrm{T}}\mathbf{L}\mathbf{B}) \\ &\text{s.t. } \mathbf{P} = \mathbf{B},\ \mathbf{B}^{\mathrm{T}}\mathbf{B} = \mathbf{I}_k \end{aligned} \tag{14}$$

where $\mathbf{P}$ is auxiliary variable. The augmented Lagrangian function for (14) is:

$$\begin{aligned} \mathscr{L}(\mathbf{B}, \mathbf{P}) = &\left\|\mathcal{X} - [\![\mathbf{B}, \mathbf{P}, \mathbf{A}]\!]\right\|_F^2 + \mathrm{Tr}\left(\mathbf{U}^{\mathrm{T}}(\mathbf{P} - \mathbf{B})\right) \\ &- \frac{\mu}{2}\|\mathbf{P} - \mathbf{B}\|_F^2 + \alpha \mathrm{Tr}\left(\mathbf{B}^{\mathrm{T}}\mathbf{L}\mathbf{B}\right) \end{aligned} \tag{15}$$

where $\mathbf{U} \in \mathbf{R}^{n \times k}$ are Lagrange multipliers, and μ is the penalty parameter. Then the objective function with respect to $\mathbf{B}$ can be derived as:

$$\begin{aligned} &\min_{\mathbf{B}} \left\|\mathbf{B}\mathbf{Q}^{\mathrm{T}} - \mathbf{X}_{(1)}\right\|_F^2 + \frac{\mu}{2}\left\|\mathbf{B} - \mathbf{P} - \frac{1}{\mu}\mathbf{U}\right\|_F^2 + \alpha \mathrm{Tr}\left(\mathbf{B}^{\mathrm{T}}\mathbf{L}\mathbf{B}\right) \\ &\text{s.t. } \mathbf{B}^{\mathrm{T}}\mathbf{B} = \mathbf{I}_k \end{aligned} \tag{16}$$

where $\mathbf{Q} = \mathbf{P} \odot \mathbf{A} \in \mathbf{R}^{(n*v) \times k}$ and $\mathbf{X}_{(1)} \in \mathbf{R}^{n \times (n*v)}$ is the mode-1 matricization of $\mathcal{X}$.

As such an optimization problem with orthogonal constraint has been well studied, and can be solved by a few solvers [1, 41], here we employ the solver *Algorithm 2* in [41] to solve Equation (16), which is a more efficient optimization algorithm with code publicly available. Since this algorithm requires the derivative of the objective function as one input, we obtain the derivative of Equation (16) with respect to $\mathbf{B}$:

$$\begin{aligned} \nabla_{\mathbf{B}}\mathscr{L}(\mathbf{B}) = &2\mathbf{B}\mathbf{Q}^{\mathrm{T}}\mathbf{Q} - 2\mathbf{X}_{(1)}\mathbf{Q} + \mu(\mathbf{B} - \mathbf{P}) - \\ &\mathbf{U} + \alpha\left(2\mathbf{L}\mathbf{B} + \mathbf{L}^{\mathrm{T}}\mathbf{B}\right) \end{aligned} \tag{17}$$

Then the auxiliary matrix $\mathbf{P}$ can be optimized by setting the derivative of Equation (15) with respect to $\mathbf{P}$ as 0. We have:

$$\mathbf{P} = \left(2\mathbf{X}_{(2)}\mathbf{O} + \mu\mathbf{B} - \mathbf{U}\right)\left(2\mathbf{O}^{\mathrm{T}}\mathbf{O} + \mu\mathbf{I}\right)^{-1} \tag{18}$$

where $\mathbf{O} = \mathbf{B} \odot \mathbf{A} \in \mathbf{R}^{(n*v) \times k}$ and $\mathbf{X}_{(2)} \in \mathbf{R}^{n \times (n*v)}$ is the mode-2 matricization of tensor $\mathcal{X}$.

After updating $\mathbf{B}$ and $\mathbf{P}$, we optimize the Lagrangian multipliers $\mathbf{U}$ by gradient ascent:

$$\mathbf{U} \leftarrow \mathbf{U} + \mu(\mathbf{P} - \mathbf{B}) \tag{19}$$

Note that in our experiment, we initialize μ as 10^{-6}, and set $\mu_{max} = 10^7$. Each time after $\mathbf{U}$ is updated, we adjust μ by $\mu = \min(\rho\mu, \mu_{max})$, where we set $\rho = 1.05$.

Update factor matrix A. Next, we fix $\mathbf{B}$ and optimize $\mathbf{A}$. Following Equation (12), the objective function with respect to $\mathbf{A}$ is:

$$\min_{\mathbf{A}} \left\|\mathbf{A}\mathbf{Z}^{\mathrm{T}} - \mathbf{X}_{(3)}\right\|_F^2 \tag{20}$$

where $\mathbf{Z} = \mathbf{B} \odot \mathbf{P} \in \mathbf{R}^{(n*n) \times k}$ and $\mathbf{X}_{(3)} \in \mathbf{R}^{v \times (n*n)}$ is the mode-3 matricization of $\mathcal{X}$, thus this can be solved directly.

By performing the above optimization steps iteratively until convergence, we can obtain the optimal indicator matrix $\mathbf{B}$ for the multi-view clustering stage, thus knowing which graphs are clustered together by performing k-means algorithm on the row vectors of $\mathbf{B}$. The resulted cluster information will be used for determining the weight matrix $\mathbf{W}$ in the multi-view graph embedding stage.

Now we solve the optimization problem in Equation (13) with respect to the weight matrix $\mathbf{W}$. According to the locally linear embedding approach proposed in [32], such a minimization problem with respect to vectors can be solved as a constrained least squares problem. Since the Frobenius norm for matrices can be regarded as a generalization of the l_2 norm for vectors, we can directly derive the following equation based on the analysis in [32]:

$$\left\|\mathbf{X}_i - \sum_j w_{ij}\mathbf{X}_j\right\|_F^2 = \sum_{jr} w_{ij}w_{ir}\mathbf{C}_{jr} \tag{21}$$

Algorithm 1 MCGE

Input: $\mathcal{X}, \{\mathcal{T}_1, \cdots, \mathcal{T}_n\}, c, k, \alpha, \beta$
Output: $\mathbf{B}, \mathbf{F}$

Initialize $\mathbf{B}$ s.t. $\mathbf{B}_0^{\mathrm{T}}\mathbf{B}_0 = \mathbf{I}_k$;
Initialize $\mathbf{F}_i$ for $i = 1, 2, \cdots, n$ s.t. $\mathbf{F}_{i0}^{\mathrm{T}}\mathbf{F}_{i0} = \mathbf{I}_c$;
while not converge **do**
  Compute $\mathbf{W}$ according to Equation (24);
  for $i = 1 : n$ **do**
    $t \leftarrow 0$;
    while not converge **do**
      Compute $\mathbf{F}_{i\,t+1}$ by solving Equation (8);
      $t \leftarrow t + 1$;
    end while
  end for
  Update $\mathbf{B}$ by solving Equation (14);
  Update $\mathbf{A}$ by solving Equation (20);
  Cluster $\mathbf{B}$ by k-means;
end while

where G_j and G_r are two neighbor graphs of G_i in the same cluster. $\mathbf{C}_{jr}$ is the local covariance matrix, and it can be computed by

$$\mathbf{C}_{jr} = \frac{1}{2}(M_j + M_r - m_{jr} - M_0) \tag{22}$$

where m_{jr} denotes the squared distance between the jth and rth neighbors of G_i, and we compute it based on both the original affinity matrices from v views and the graph embeddings of G_j and G_r by

$$m_{jr} = \frac{1}{2}(\frac{1}{d}\sum_{d=1}^{v}\left\|\mathbf{G}_j^{(d)} - \mathbf{G}_r^{(d)}\right\|_F^2) + \frac{1}{2}(\left\|\mathbf{F}_j - \mathbf{F}_r\right\|_F^2) \tag{23}$$

$M_j = \sum_z m_{jz}$, $M_r = \sum_z m_{rz}$ and $M_0 = \sum_{jr} m_{jr}$. Then the optimal weights can be obtained by:

$$w_{ij} = \frac{\sum_r \mathbf{C}_{jr}^{-1}}{\sum_{lz} \mathbf{C}_{lz}^{-1}} \tag{24}$$

For details about the above derivation for the solution, readers can refer to the illustrations in [32].

Once the weight matrix $\mathbf{W}$ is obtained, we can easily solve the optimization problem in Equation (8) following the same ADMM steps as shown above for solving (12). The overall optimization algorithm of MCGE is summarized in Algorithm 1.

5 EXPERIMENTS AND EVALUATION

In order to evaluate the performance of the proposed method for multi-view clustering of graphs, we test our framework on real fMRI and DTI brain network data for connectome analysis and compare with a few of state-of-the-art multi-view clustering methods.

5.1 Data Collection and Preprocessing

In this work, we use two real datasets as follows:

- *Human Immunodeficiency Virus Infection (HIV)*: This dataset is collected from the Chicago Early HIV Infection Study at Northwestern University[30]. This clinical study involves 77 subjects, 56 of which are early HIV patients (positive) and the other 21 subjects are seronegative controls (negative). These two groups of subjects do not differ in demographic characteristics such as age, gender, racial composition and education level. This dataset contains both the functional magnetic resonance imaging (fMRI) and diffusion tensor imaging (DTI) for each subject, from which we can construct the fMRI and DTI brain networks.
- *Bipolar*: This dataset consists of the resting-state fMRI and DTI image data of 52 bipolar I subjects who are in euthymia and 45 healthy controls with matched age and gender [9, 25].

We perform preprocessing on the HIV dataset using the standard process as illustrated in [8]. First, we use the DPARSF toolbox[1] to process the fMRI data. We realign the images to the first volume, do the slice timing correction and normalization, and then use an 8-mm Gaussian kernel to smooth the image spatially. The band-pass filtering (0.01-0.08 Hz) and linear trend removing of the time series are also performed. We focus on the 116 anatomical volumes of interest (AVOI), each of which represents a specific brain region, and extract a sequence of responds from them. Finally, we construct a brain network with the 90 cerebral regions. Each node in the graph represents a brain region, and links are created based on the correlations between different brain regions. For the DTI data, we use FSL toolbox[2] for the preprocessing and then construct the brain networks. The preprocessing includes distortion correction, noise filtering, repetitive sampling from the distributions of principal diffusion directions for each voxel. We parcellate the DTI images into the 90 regions same with fMRI via the propagation of the Automated Anatomical Labeling (AAL) on each DTI image [36].

For the Bipolar dataset, the brain networks were constructed using the CONN[3] toolbox [42]. The raw EPI images were first realigned and co-registered, after which we perform the normalization and smoothing. Then the confound effects from motion artifact, white matter, and CSF were regressed out of the signal. Finally, the brain networks were derived using the pairwise signal correlations based on the 82 labeled Freesurfer-generated cortical/subcortical gray matter regions.

5.2 Baselines and Metrics

We compare our MCGE framework with six other baseline methods for the multi-view clustering task on brain networks. To the best of our knowledge, our proposed framework is the first work that jointly performs multi-view graph embedding and multi-view clustering of graph instances. Therefore, for the evaluation, we apply the following state-of-the-art multi-view clustering methods and adapt them to perform the multi-view clustering task here.

- **SingleBest** applies spectral clustering on each single view and reports the best performance among them.
- **SEC** is a single view spectral embedding clustering framework proposed in [28]. It imposes a linearity regularization on the spectral clustering model and uses both local and global discriminative information for the embedding.
- **CoRegSc** is the co-regularized based multi-view spectral clustering framework proposed in [19]. The centroid based approach is applied for the multi-view clustering task.
- **MultiNMF** is the multi-view clustering method based on joint nonnegative matrix factorization proposed by [21]. It aims to search for a factorization that gives compatible clustering solutions across multiple views.

[1] http://rfmri.org/DPARSF.
[2] http://fsl.fmrib.ox.ac.uk/fsl/fslwiki.
[3] http://www.nitrc.org/projects/conn

Table 2: Results on HIV dataset (mean ± std).

Methods	*Accuracy*	*NMI*
SingleBest	0.561 ± 0.010	0.104 ± 0.007
SEC	0.523 ± 0.012	0.092 ± 0.011
AMGL	0.563 ± 0.002	0.132 ± 0.008
SCMV-3DT	0.576 ± 0.013	0.123 ± 0.019
MultiNMF	0.613 ± 0.016	0.197 ± 0.021
CoRegSc	0.626 ± 0.020	0.254 ± 0.013
MCGE	**0.682 ± 0.019**	**0.390 ± 0.015**

Table 3: Results on Bipolar dataset (mean ± std).

Methods	*Accuracy*	*NMI*
SingleBest	0.553 ± 0.012	0.098 ± 0.006
SEC	0.536 ± 0.012	0.103 ± 0.009
AMGL	0.558 ± 0.026	0.101 ± 0.012
SCMV-3DT	0.585 ± 0.009	0.132 ± 0.010
MultiNMF	0.642 ± 0.011	0.192 ± 0.015
CoRegSc	0.619 ± 0.024	0.170 ± 0.008
MCGE	**0.703 ± 0.013**	**0.264 ± 0.012**

- **AMGL** is a recently proposed multi-view spectral learning framework [27] that can automatically learn an optimal weight for each graph without introducing additive parameters.
- **SCMV-3DT** is a tensor based multi-view clustering method recently proposed in [46]. It uses t-product in third-order tensor space, and represents multi-view data by a t-linear combination with sparse and low-rank penalty based on the circular convolution.
- **MCGE** is the proposed multi-view clustering framework in this paper, which jointly performs multi-view graph embedding and multi-view clustering of the graph instances.

There are three main parameters in our model, which include the α in objective function (12), the β in objective function (8), and the dimension c of the row vectors in the graph embeddings. We apply the grid search to find the optimal values for the parameters. For details, we do grid search for α and β in $\{10^{-4}, 10^{-3}, \cdots 10^{4}\}$, and the optimal c is selected by the grid search from $\{2, 3, \cdots, 12\}$. For evaluation, since there are two possible labels of the brain network instances in both of the two datasets, we set the number of clusters k to be 2, and test how well our method can group the brain networks of subjects with disorders and those of normal controls into two different clusters.

For fair comparisons of the baseline methods, we employ Litekmeans [6] for all the k-means clustering step if it is needed in the implementation of the six methods listed above. We repeat clustering for 20 times with random initialization as k-means depends on initialization.

To evaluate the quality of the clusters produced by different approaches, we use *Accuracy* and *Normalized Mutual Information (NMI)* as the evaluation metrics. For each experiment, we repeat 50 times and report the mean value along with standard deviation (std) as the results.

5.3 Performance Evaluations

5.3.1 Clustering Accuracy and NMI. As shown in Table 2 and Table 3, our MCGE framework performs the best in the multi-view clustering task on both of the two datasets in terms of accuracy and NMI. Among the seven methods, the first two methods are single view clustering methods, both of which achieve lower accuracy and NMI compared with the multi-view methods. In particular, the lowest accuracy is from SEC, which is a single view clustering method applied here by concatenating the features of all the views. Although the SEC method considers both global structure and local structure of graphs, it does not distinguish the features from different views, which leads to a poor performance in the multi-view clustering. The SingleBest achieves its best performance on the fMRI brain networks for both datasets, which means that the fMRI data provide more discriminative information for the SingleBest method. By comparing SingleBest with SEC, we can find that if the multiple views are combined improperly, it may perform even worse than only using information from a single view.

Among the multi-view clustering methods, CoRegSc and MultiNMF have quite good performance, though not as good as the proposed MCGE method. This is mainly because that they consider the interactions between different views via joint modeling with the multiple views, while the other two multi-view methods do not. Comparatively, CoRegSc achieves slightly better results than the MultiNMF method on HIV dataset and vice versa on the Bipolar dataset. Compared to the proposed MCGE method, the common property of the other four multi-view clustering methods is that the features they learn for each view are based on vector representations. However, for graph instances, the structural information could barely be preserved by such vector representations, which could be the underlying reason of why these methods could not outperform our MCGE method. Moreover, by using tensor technique to model the multi-view graph-graph affinity as illustrated in Equation (9), MCGE can not only encode the latent interaction across different views, but also capture the graph-specific features through the graph kernels. From Table 2 and Table 3, we can see that, as another tensor-based multi-view method, the SCMV-3DT does not achieve compatible results to MCGE. The reason behind this might be that although SCMV-3DT models the data into third-order tensor, it does not consider the local structure of graphs, making it less effective for the multi-view clustering of graphs.

5.3.2 MCGE for Connectome Analysis. To evaluate the effectiveness of the proposed MCGE framework for connectome analysis, we investigate this approach for capturing the inner structure of connectomes in analysis of brain alterations induced by HIV infection and Bipolar affective disorder, respectively.

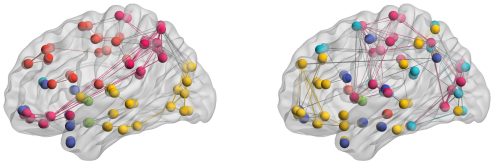
HIV is associated with heterogeneous changes in the brain and in cognitive function [39]. In many CNS(Central Nervous System) disorders, etiology is unknown. In contrast, HIV involves a known viral etiology. Therefore it is possible to study the brain in the early stages of injury. Studies of early HIV infection have found alterations in both structural and functional connectivity [39]. Moreover, a hallmark of HIV is neuroinflammation, which is a common characteristic of neurological injury from diverse causes, including traumatic, ischemic, developmental and neurodegenerative brain disorders. Since HIV infection is broadly relevant to many other

neurological disorders, it represents an ideal model for evaluating the sensitivity of new frameworks for neuroimaging analysis.

We apply the proposed MCGE framework on the multi-view brain networks of the HIV dataset and obtain the clustering results as well as the multi-view graph embedding for each brain network. We further employ k-means algorithm (with $k = 6$) on the row vectors of the multi-view graph embedding for each brain network, and obtain the clustering relationship of their inner nodes, *i.e.*, the brain regions. Figure 4 shows an example of the resulting brain region clustering map of a normal control and that of an HIV patient. In this figure, each node represents a brain region, and each edge indicates the correlation between two brain regions. Nodes of the same color represent the brain regions that are grouped into the same cluster by MCGE. As we can see from Figure 4, the clustering pattern of the HIV patient is quite different from the normal control. Nodes of the normal brain network are well grouped into several clusters, while nodes in the HIV brain network are less coherent. In addition, for the normal control, edges within each cluster are much more intense than the edges across different clusters. For example, in Figure 4(a), the pink nodes in the lower left and the pink nodes in the upper right are strongly connected with each other. While in Figure 4(b), the corresponding nodes in the lower left, which are mostly marked in yellow, have very few connections with those yellow nodes in the upper right. By looking into the connections, we can find that for the normal control, there are several pink nodes in the center of the brain which bridge the lower left part and the upper right part, while these intermediate nodes in the HIV brain are clustered in blue or pink instead of the same color (yellow) as the lower left part and the upper right part. This implies that the intermediate regions are probably injured so that they are no longer the bridges (or hubs) across other related regions. Some studies in neuroscience [12] show that the highly-interconnected hub nodes are biologically costly due to higher blood flow or connection distances, and thus tend to be more sensitive to injury. Our observations in Figure 4 potentially reflect this evidence.

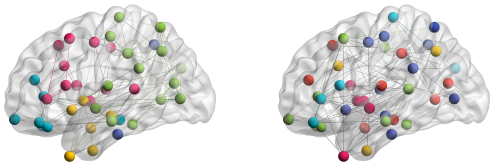
Then we apply the MCGE framework on the Bipolar dataset with the same steps as illustrated above for HIV dataset. The visualized results of a normal control and a bipolar subject are shown in Figure 5. Similarly to the observations above, as we can see from Figure 5, the cluster information of normal control is quite different from the bipolar subject. The connectomes of the normal control are well organized, while the corresponding nodes in the brain network of the bipolar subject spread out irregularly across different clusters. We can also find that for normal control, edges within each cluster are much more intense than the edges across different clusters, while this is less the case for bipolar subject. The reason behind this observation is probably that the collaborative activities of different brain regions of the bipolar subject are not organized in a proper order as those of normal controls are.

These findings indicate that our proposed MCGE framework can distinguish brain alterations in neurological disorders from healthy controls. It also yields new information and insights concerning network perturbations in brain injury and neuroinflammation for further investigation and interpretation.



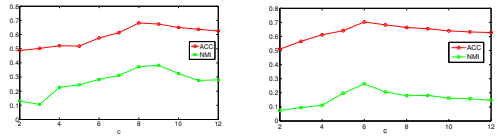
(a) normal control (b) HIV patient

Figure 4: Comparison of the connectomes captured from the brain networks of a normal control and an HIV patient



(a) normal control (b) bipolar subject

Figure 5: Comparison of the connectomes captured from the brain networks of a normal control and a bipolar subject

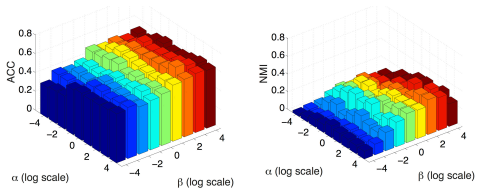


(a) *HIV* (b) *Bipolar*

Figure 6: *Accuracy* and *NMI* with different c

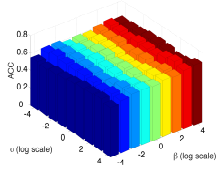
5.4 Parameter Sensitivity Analysis

In this section, we study the sensitivity of the proposed MCGE framework to the three parameters α, β, and c, and explore how the different values for parameters would affect the performance of MCGE in the multi-view clustering. We first look into the parameter c, which is the dimension of the row vectors in graph embedding. Figure 6 shows the multi-view clustering performance of MCGE on the two datasets with the c value varying from 2 to 12. From the figure, we can see that the value for c affects the performance of MCGE in both accuracy and NMI. The highest accuracy is achieved when c equals to 8 for HIV dataset and the best NMI occurs at 9. For Bipolar dataset, both the accuracy and NMI reach the peak when c equals to 6. The changing of accuracy and NMI with different c values has similar trend on the two datasets. With the increase of the c value, the performance first keeps rising up until it reaches the peak, and then it starts to decline. This changing trend is reasonable as when the dimension of graph embedding is too small, it could not encode enough local structure information of the graph, resulting in poor performance for the clustering. When the dimension of graph embedding is set to be a large number, it may include much redundant information, making it less discriminative for the clustering task.

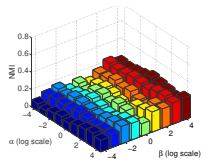


(a) *Accuracy on HIV*

(b) *NMI on HIV*



(c) *Accuracy on Bipolar*



(d) *NMI on Bipolar*

Figure 7: *Accuracy* and *NMI* with different α, β

Now we evaluate the sensitivity of MCGE to α and β. As illustrated in Equation (12), α is the weight parameter which determines the extent that the local embedding structure is utilized for the multi-view clustering task. The higher the value for α, the more emphasis we put on the graph embeddings for multi-view clustering modeling. Similarly, the parameter β balances how much influence the embeddings of neighbor graphs would have on the multi-view graph embedding of each graph. For the evaluation, we set c to be 8 and run the MCGE framework with different values of α and β. The clustering accuracy and NMI achieved at different values of parameters for the two datasets are shown in Figure 7(a), Figure 7(b), Figure 7(c) and Figure 7(d), respectively. As we can see from the figures, MCGE achieves different levels of accuracy and NMI when the values of α and β vary. The highest accuracy on HIV dataset is achieved when $\alpha = 10^3$, and $\beta = 10^2$, while the best NMI on HIV is achieved at $\alpha = 10^3$, and $\beta = 10^3$. On Bipolar dataset, both the highest accuracy and the best NMI are achieved when $\alpha = 10^3$, and $\beta = 10^3$. Notably, when the value for α is too small, both the accuracy and NMI achieved by MCGE are quite low, and the same situation holds for β. This is mainly because that if we set a small value to α, little information of graph embeddings would be used for the multi-view clustering stage. Similarly, when β is too small, the graph embeddings of neighbor graphs would hardly influence the multi-view graph embedding stage of each graph. On the other hand, when α and β are set to be large values, the performance drops as well, as the influence imposed on those parts is too much. Therefore, finding an optimal combination of these parameter values is very important when applying MCGE framework for multi-view clustering.

6 RELATED WORK

Our work relates to several branches of studies, which include multi-view clustering, graph embedding and connectome analysis.

Multi-view clustering is a clustering strategy for analyzing data with multiple views [3] and it has been widely studied and applied in various domains [22, 33, 34]. For example, the Canonical Correlation Analysis (CCA) based methods focus on constructing projections using multiple views[3]. In [10], a CCA based method is proposed and applied for audio-visual speaker clustering and hierarchical Wikipedia document clustering. Another main category of algorithms aim to integrate multiple views in the clustering process directly by optimizing the loss functions[3]. A typical work from this category is the co-regularized multi-view spectral clustering method proposed by [19], which is also a baseline method used in our experiment. It performs multi-view clustering by co-regularizing the clustering hypotheses. In addition, matrix factorization based methods also form a category of multi-view clustering methods[18, 21], which use constraints to push multiple views towards consensus.

Graph embedding is a hot research topic in graph mining. The goal of graph embedding is to find low-dimensional representations of nodes in graphs that can preserve the important structure and properties of graphs [45]. It has drawn great interest from the data mining community, and has been extensively studied for various kinds of applications. In [26], a new graph embedding algorithm is proposed based on Laplacian type operator on manifold, and it is applied for recovering the geometry of data and extending a function on new data points. Recently, a high-order proximity preserved embedding method is proposed in [29]. They first derive a general formulation that covers multiple popular high-order proximity measurements, and then propose a scalable embedding algorithm to approximate the high-order proximity measurements.

Connectome analysis is a prominent emphasis area in the field of medical data mining. The "connectome", refers to the vast connectivity of neural systems at different levels involving both global and local structure information of the connections [17]. Connectome analysis has been the focus of intense investigation owing to the tremendous potential to provide more comprehensive understanding of normal brain function and to yield new insights concerning many different brain disorders [7, 25, 35]. Most connectome analyses, however, aim to learn the structure from brain networks based on an individual neuroimaging modality [9, 15, 16, 20]. For example, in [9], the identification of discriminative subgraph patterns is studied on fMRI brain networks for bipolar affective disorder analysis. In [23], a multi-graph clustering method is proposed based on interior-node clustering for connectome analysis in fMRI resting-state networks. Although some recent work [5] use multi-view brain networks in connectome analysis, they focus on the group-wise functional community detection problem instead of doing multi-view clustering of the subjects. Here, we apply the proposed graph embedding based approach to facilitate the multi-view clustering of multiple brain networks simultaneously, thus providing a more comprehensive strategy for further neurological disorder identification.

7 CONCLUSION

In this paper, we present MCGE, a Multi-view Clustering framework with Graph Embedding, to solve multi-view clustering problem on graph instances. MCGE first models the multi-view graph data as tensors and then learns the multi-view graph embeddings via tensor factorization. We further incorporate multi-view graph embedding into an iterative multi-view clustering framework, jointly

performing multi-view clustering and graph embedding simultaneously. The results of multi-view clustering are used to refine the multi-view graph embeddings, in turn, the updated multi-view graph embedding results are used to improve the multi-view clustering. By updating the clustering results and graph embeddings iteratively, the proposed MCGE framework will result in a better multi-view clustering solution. We apply our MCGE framework for unsupervised multi-view connectome analysis on HIV-induced brain alterations and bipolar affective disorder. Extensive experimental results on real multi-view HIV brain network data and Bipolar brain network data show the effectiveness of MCGE for multi-view clustering in connectome analysis.

ACKNOWLEDGMENTS

This work is supported in part by NSF through grants IIS-1526499, and CNS-1626432, and NSFC 61672313, and NSFC 61503253.

REFERENCES

[1] P-A Absil, Robert Mahony, and Rodolphe Sepulchre. 2009. *Optimization algorithms on matrix manifolds*. Princeton University Press.
[2] Mikhail Belkin and Partha Niyogi. 2001. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *NIPS*.
[3] Steffen Bickel and Tobias Scheffer. 2004. Multi-View Clustering. In *ICDM*.
[4] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. 2011. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning* 3, 1 (2011), 1–122.
[5] Nathan D Cahill, Harmeet Singh, Chao Zhang, Daryl A Corcoran, Alison M Prengaman, Paul S Wenger, John F Hamilton, Peter Bajorski, and Andrew M Michael. 2016. Multiple-View Spectral Clustering for Group-wise Functional Community Detection. *arXiv preprint arXiv:1611.06981* (2016).
[6] D Cai. 2011. Litekmeans: the fastest matlab implementation of kmeans. *Software available at: http://www. zjucadcg. cn/dengcai/Data/Clustering. html* (2011).
[7] Bokai Cao, Lifang He, Xiaokai Wei, Mengqi Xing, Philip S. Yu, Heide Klumpp, and Alex D. Leow. 2017. t-BNE: Tensor-based Brain Network Embedding. In *SDM*.
[8] Bokai Cao, Xiangnan Kong, Jingyuan Zhang, S Yu Philip, and Ann B Ragin. 2015. Identifying HIV-induced subgraph patterns in brain networks with side information. *Brain informatics* 2, 4 (2015), 211–223.
[9] Bokai Cao, Liang Zhan, Xiangnan Kong, Philip S Yu, Nathalie Vizueta, Lori L Altshuler, and Alex D Leow. 2015. Identification of discriminative subgraph patterns in fMRI brain networks in bipolar affective disorder. In *BIH*.
[10] Kamalika Chaudhuri, Sham M Kakade, Karen Livescu, and Karthik Sridharan. 2009. Multi-view clustering via canonical correlation analysis. In *ICML*.
[11] Rodrigo Cilla Ugarte. 2012. Action recognition in visual sensor networks: a data fusion perspective. (2012).
[12] Nicolas A Crossley, Andrea Mechelli, Jessica Scott, Francesco Carletti, Peter T Fox, Philip McGuire, and Edward T Bullmore. 2014. The hubs of the human connectome are generally implicated in the anatomy of brain disorders. *Brain* 137, 8 (2014), 2382–2395.
[13] Yun Fu and Yunqian Ma. 2012. *Graph embedding for pattern analysis*. Springer Science & Business Media.
[14] Jing Gao, Nan Du, Wei Fan, Deepak Turaga, Srinivasan Parthasarathy, and Jiawei Han. 2013. A multi-graph spectral framework for mining multi-source anomalies. In *Graph Embedding for Pattern Analysis*. Springer, 205–227.
[15] Lifang He, Xiangnan Kong, Philip S Yu, Xiaowei Yang, Ann B Ragin, and Zhifeng Hao. 2014. Dusk: A dual structure-preserving kernel for supervised tensor learning with applications to neuroimages. In *SDM*.
[16] Lifang He, Chun-Ta Lu, Guixiang Ma, Shen Wang, Linlin Shen, S Yu Philip, and Ann B Ragin. 2017. Kernelized support tensor machines. In *ICML*.
[17] Marcus Kaiser. 2011. A tutorial in connectome analysis: topological and spatial features of brain networks. *Neuroimage* 57, 3 (2011), 892–907.
[18] Mahdi M Kalayeh, Haroon Idrees, and Mubarak Shah. 2014. Nmf-knn: Image annotation using weighted multi-view non-negative matrix factorization. In *CVPR*.
[19] Abhishek Kumar, Piyush Rai, and Hal Daume. 2011. Co-regularized multi-view spectral clustering. In *NIPS*.
[20] Chia-Tung Kuo, Xiang Wang, Peter Walker, Owen Carmichael, Jieping Ye, and Ian Davidson. 2015. Unified and contrasting cuts in multiple graphs: application to medical imaging segmentation. In *SIGKDD*.
[21] Jialu Liu, Chi Wang, Jing Gao, and Jiawei Han. 2013. Multi-view clustering via joint nonnegative matrix factorization. In *SDM*.
[22] Chun-Ta Lu, Lifang He, Weixiang Shao, Bokai Cao, and Philip S Yu. 2017. Multilinear Factorization Machines for Multi-Task Multi-View Learning. In *WSDM*.
[23] Guixiang Ma, Lifang He, Bokai Cao, Jiawei Zhang, S Yu Philip, and Ann B Ragin. 2016. Multi-graph Clustering Based on Interior-Node Topology with Applications to Brain Networks. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 476–492.
[24] Guixiang Ma, Lifang He, Chun-Ta Lu, Philip S Yu, Linlin Shen, and Ann B Ragin. 2016. Spatio-temporal tensor analysis for whole-brain fMRI classification. In *Proceedings of the 2016 SIAM International Conference on Data Mining*. SIAM, 819–827.
[25] Guixiang Ma, Chun-Ta Lu, Lifang He, S Yu Philip, and B Ragin Ann. 2017. Multi-view Graph Embedding with Hub Detection for Brain Network Analysis. In *ICDM*.
[26] Saman Mousazadeh and Israel Cohen. 2015. Embedding and function extension on directed graph. *Signal Processing* 111 (2015), 137–149.
[27] Feiping Nie, Jing Li, Xuelong Li, and others. 2016. Parameter-Free Auto-Weighted Multiple Graph Learning: A Framework for Multiview Clustering and Semi-Supervised Classification. International Joint Conferences on Artificial Intelligence.
[28] Feiping Nie, Zinan Zeng, Ivor W Tsang, Dong Xu, and Changshui Zhang. 2011. Spectral embedded clustering: A framework for in-sample and out-of-sample spectral clustering. *IEEE Trans on Neural Networks* 22, 11 (2011), 1796–1808.
[29] M Ou, Peng Cui, Jian Pei, Z Zhang, and W Zhu. 2016. Asymmetric transitivity preserving graph embedding. In *SIGKDD*.
[30] Ann B Ragin, Hongyan Du, Renee Ochs, Ying Wu, Christina L Sammet, Alfred Shoukry, and Leon G Epstein. 2012. Structural brain alterations can be detected early in HIV infection. *Neurology* 79, 24 (2012), 2328–2334.
[31] Sam T Roweis and Lawrence K Saul. 2000. Nonlinear dimensionality reduction by locally linear embedding. *Science* 290, 5500 (2000), 2323–2326.
[32] Lawrence K Saul and Sam T Roweis. 2000. An introduction to locally linear embedding. *http://www. cs. toronto. edu/~ roweis/lle/publications. html* (2000).
[33] Weixiang Shao, Lifang He, Chun-Ta Lu, Xiaokai Wei, and S Yu Philip. 2016. Online unsupervised multi-view feature selection. In *ICDM*.
[34] Weixiang Shao, Lifang He, and S Yu Philip. 2015. Clustering on multi-source incomplete data via tensor modeling and factorization. In *PAKDD*.
[35] Olaf Sporns, Giulio Tononi, and Rolf Kötter. 2005. The human connectome: a structural description of the human brain. *PLoS Comput Biol* 1, 4 (2005), e42.
[36] Nathalie Tzourio-Mazoyer, Brigitte Landeau, Dimitri Papathanassiou, Fabrice Crivello, Olivier Etard, Nicolas Delcroix, Bernard Mazoyer, and Marc Joliot. 2002. Automated anatomical labeling of activations in SPM using a macroscopic anatomical parcellation of the MNI MRI single-subject brain. *Neuroimage* 15, 1 (2002), 273–289.
[37] S Vichy N Vishwanathan, Nicol N Schraudolph, Risi Kondor, and Karsten M Borgwardt. 2010. Graph kernels. *Journal of Machine Learning Research* 11, Apr (2010), 1201–1242.
[38] Ulrike Von Luxburg. 2007. A tutorial on spectral clustering. *Statistics and computing* 17, 4 (2007), 395–416.
[39] Xue Wang, Paul Foryt, Renee Ochs, Jae-Hoon Chung, Ying Wu, Todd Parrish, and Ann B Ragin. 2011. Abnormalities in resting-state functional connectivity in early human immunodeficiency virus infection. *Brain connectivity* 1, 3 (2011), 207–217.
[40] Yichen Wang, Robert Chen, Joydeep Ghosh, Joshua C Denny, Abel Kho, You Chen, Bradley A Malin, and Jimeng Sun. 2015. Rubik: Knowledge guided tensor factorization and completion for health data analytics. In *SIGKDD*.
[41] Zaiwen Wen and Wotao Yin. 2013. A feasible method for optimization with orthogonality constraints. *Mathematical Programming* 142, 1-2 (2013), 397–434.
[42] Susan Whitfield-Gabrieli and Alfonso Nieto-Castanon. 2012. Conn: a functional connectivity toolbox for correlated and anticorrelated brain networks. *Brain connectivity* 2, 3 (2012), 125–141.
[43] Tian Xia, Dacheng Tao, Tao Mei, and Yongdong Zhang. 2010. Multiview spectral embedding. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 40, 6 (2010), 1438–1446.
[44] Bo Xie, Yang Mu, Dacheng Tao, and Kaiqi Huang. 2011. m-SNE: Multiview stochastic neighbor embedding. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 41, 4 (2011), 1088–1096.
[45] Shuicheng Yan, Dong Xu, Benyu Zhang, Hong-Jiang Zhang, Qiang Yang, and Stephen Lin. 2007. Graph embedding and extensions: a general framework for dimensionality reduction. *IEEE transactions on pattern analysis and machine intelligence* 29, 1 (2007), 40–51.
[46] Ming Yin, Shengli Xie, Yi Guo, and others. 2016. Low-rank Multi-view Clustering in Third-Order Tensor Space. *arXiv preprint arXiv:1608.08336* (2016).
[47] Jingyuan Zhang, Bokai Cao, Sihong Xie, Chun-Ta Lu, Philip S Yu, and Ann B Ragin. 2016. Identifying connectivity patterns for brain diseases via multi-side-view guided deep architectures. In *SDM*.
[48] Lefei Zhang, Qian Zhang, Liangpei Zhang, Dacheng Tao, Xin Huang, and Bo Du. 2015. Ensemble manifold regularized sparse low-rank approximation for multiview feature embedding. *Pattern Recognition* 48, 10 (2015), 3102–3112.