

iMGC: Interactive Multiple Graph Clustering with Constrained Laplacian Rank

Zhiguang Zhou¹ , Haoxuan Wang¹ , Ling Sun¹ , Yuhua Liu² , Xiang Zhang¹ , Yigang Wang² , and Wei Chen³ 

¹School of Information, Zhejiang University of Finance & Economics, China

²School of Meida and Design, Hangzhou Dianzi University, China

³State Key Lab of CAD and CG, Zhejiang University, China

Abstract

Numerous graph clustering methods have been proposed to explore aggregation structures across multiple graphs. In these methods, single-graph features are merely considered or multi-graph features are simply weighted, which are insufficient for the construction of reasonable multiple graph clustering features, since the association information between pairwise graphs is ignored and the varied local correlations might influence the clustering preference. Thus, we propose an interactive multiple graph clustering model, iMGC, in this paper, to achieve reasonable multiple graph clustering features, which cannot only express multiple relationships, but also preserve associations of nodes across multiple graphs. First, a unified graph matrix is constructed with the combination of structural differences quantified by graph representation learning, which is further optimized by minimizing the difference of structural characteristics between it and each single graph matrix. Thus, multiple relationships are well integrated and expressed, while the varied local correlations within different graphs are also balanced in the unified graph matrix. Then, a constrained Laplacian rank is applied on the unified graph matrix to generate the unified clustering result directly, which is able to preserve association features across multiple graphs. Furthermore, we provide a set of visualization and interaction interfaces, enabling users to intuitively optimize and evaluate the multiple graph clustering features, and interactively explore the multiple graphs. Case studies and quantitative comparisons based on real-world datasets have demonstrated the effectiveness of iMGC in the clustering performance from various perspectives and exploration of multiple graphs.

CCS Concepts

- Human-centered computing → Information visualization;

1. Introduction

Graph clustering is a commonly-used method for the exploration of large-scale networks, which always partitions graph into several densely connected components based on node similarity. It is of great significance for information mining and feature exploration from complex networks, thus has been widely applied in many application areas ranging from community networks [LG19] and recommendation system [BSZG18] to biological science [HYZ*19] and 3D pose estimation [CGL*19].

With the rapid development of sensors, relations among objects always present diversified expressions and form complex multiple network relationships. For example, individuals can be related in a variety of social networks, such as friendships, commodity transactions and online communications. Traditional graph clustering methods are hardly suitable for the aggregation of multiple graph structures, because they merely consider single-graph features or simply weighted multi-graph features. As shown in Figure 1d, a set

of pink points are clustered in network ElectrJ due to their close relationship, which are quite irrelevant in the network MonoSyn (Figure 1f) and network PolySyn (Figure 1h). It means that we will lose the relationship of interest if we merely consider the last two networks. Similarly, we weight the connections of points across multiple graphs and conduct graph clustering on the syncretic graph, while the relationship of interest in network ElectrJ is largely weakened and lose its original associations as shown in Figure 1e. Therefore, it is quite in demand for designing a graph clustering method to achieve reasonable clustering features with the associations of nodes considered comprehensively across multiple graphs.

To conduct graph clustering with associations of nodes comprehensively considered across multiple graphs, we are facing three main challenges: **CH1**. It is a tough task to capture and quantify structural characteristics between pairwise graphs, especially those varied local correlations across different graphs. **CH2**. It is also difficult enough to achieve reasonable clustering features to consider the associations between pairwise graphs and balance the repre-

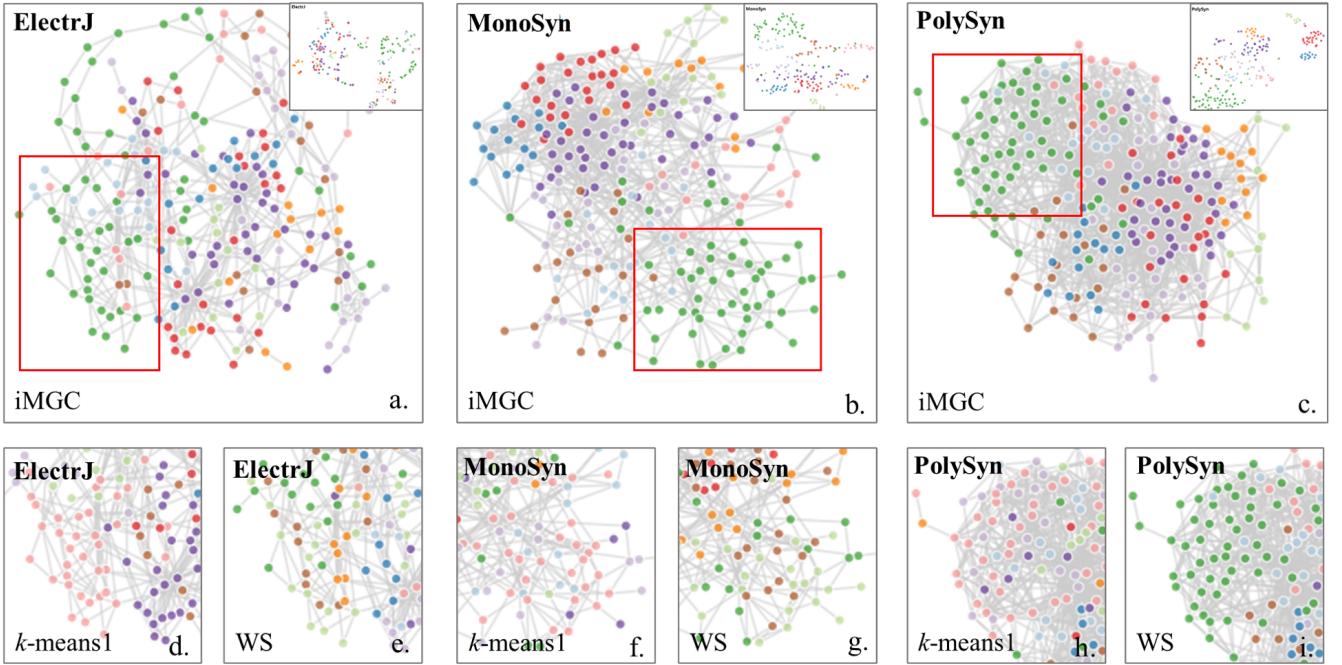


Figure 1: A case study is conducted on a real-world multiple graph CELE, which consists of three graphs (ElectrJ, MonoSyn and PolySyn) sharing common nodes and different edges. A group of clustering results based on different methods are listed (each column presents a graph from left to right): (d, f, h) k-means 1, (e, g, i) WS, (a, b, c) iMGC.

sentation of structural characteristics within different graphs. **CH3.** It is another challenge to design quantitative and visual metrics to evaluate multiple graph clustering results from a variety of perspectives, such as the preservation of associations and user preferences.

To tackle the above challenges, we propose a multiple graph clustering model to achieve reasonable multiple graph clustering features, with the multiple relations and associations of nodes across multiple graphs considered comprehensively. First, we construct a unified graph matrix to combine the structure difference of each graph measured by node2vec, and optimize the unified graph matrix by minimizing the difference of structural characteristics between it and each graph matrix. Thus, the unified graph matrix will balance the varied local correlations within different graphs as far as possible (**CH1**). Then, we take advantage of a constrained Laplacian rank to generate the multigraph clustering features from the unified matrix, in which the associations between pairwise graphs are well preserved (**CH2**). Also, we propose a group of quantitative and visual metrics to evaluate the validity of our clustering method in multiple relationship integration and association feature preservation. A series of interactions are further provided to support the interactive optimization of clustering features based on user preferences (**CH3**). At last, a multiple graph clustering framework is implemented to integrate graph clustering models, visual metrics and a series of interactions, allowing users to optimize, evaluate and explore the clustering features quickly, intuitively and in real time. The effectiveness and usefulness of our system are demonstrated with case studies and quantitative comparisons based on real-world datasets. The major contributions of this paper are summarized as follows:

- A unified graph matrix is constructed to combine structural characteristics of nodes quantified by node2vec. Compared with the other traditional clustering methods, node similarity in the unified graph matrix is derived from graph representation learning space, which largely facilitates the balance of varied local correlations within different graphs.
- A multiple graph clustering method is proposed based on the unified graph matrix, which utilizes the constrained Laplacian rank to minimize the weights of cut edges in the process of node aggregations, so that the association features of nodes across multiple graphs are largely preserved in the clustering results.
- An interactive multiple graph clustering framework is implemented in this paper, with a rich set of visual cues and metrics integrated, enabling users to easily obtain, evaluate and optimize the clustering results, and then conduct in-depth exploration on complex multiple graph datasets.

The rest of this paper is structured as follows. Section 2 introduces related work. Section 3 summarizes the requirement tasks and presents the system overview. Section 4 depicts the proposed multiple graph clustering model in detail. The visual designs and interactions are described in Section 5. Quantitative comparisons and case studies based on the real-world datasets are presented in Section 6. We draw conclusions from this work and summarize the future work in Section 7.

2. Related Work

In this paper, related work is classified into three categories, including graph analysis, graph clustering and graph visualization.

2.1. Graph Analysis

Recently, there are three main state-of-art graph data structures: Compressed Sparse Columns [Saa90], Compressed Sparse Rows [Saa90, SVN17], and Coordinate list [Saa90]. Graph analysis aims to deeply understand the structures of relationships in graphs. For example, node degree, degree centrality, proximity centrality, intermediate centrality, shortest path, etc. are the basic metrics for solving many graph analysis problems [KRUT09]. To mine the rich structural information of graphs, a lot of graph embedding methods have been proposed to represent the graph nodes in vector space preserving the graph structural information. Three categories are covered: (1) Factorization-based methods factorize the matrix representing the connections between nodes to obtain the embedding, such as LLE [RS00], Laplacian Eigenmaps [BN02]. (2) Random walk-based methods generate random walks across nodes, and then apply neural language models to get embedding, such as Deepwalk [PAS14], node2vec [GL16]. (3) Deep learning-based methods combine the neighborhood node information with the center node, such as SDNE [WCZ16], DNGR [ZWC20]. In addition, graph data mining approaches have been developed to exploit the graph structures. For example, link prediction aims at predicting whether two nodes in a network have a link based on the definition of structural similarity measures between unconnected node pairs [ZC18]. Graph clustering is another important graph mining task that aims to reveal the inherent organizational structural of graph by grouping similar nodes together.

2.2. Graph Clustering

A large variety of algorithms have been designed for graph clustering, which can be roughly classified into the following five categories. (1) In the centroid-based clustering strategies, each cluster is represented with a central vector and nodes are assigned to the clusters based on proximity, such as k -means [HW79], Fuzzy c-mean [OIO*19]. (2) Connectivity-based methods take a linkage criterion and distance metric to split clusters, such as Hierarchical and Agglomerative methods [OIO*19]. (3) Density-based methods define a cluster as a maximal set of density-connected points, such as DBSCAN [RSJM10a], OPTICS [RSJM10b]. (4) Probabilistic-based methods use probability distributions to determine which cluster the points belong to, such as DeepWalk [PAS14], Gaussian Mixture Models [RG02]. (5) Matrix factorization-based methods aim to obtain low embedding by decomposing the adjacency matrix, such as modularized nonnegative matrix factorization [GCXW20]. These methods typically conduct graph clustering based on similarity matrix, which brings additional Probably Approximately Correct bounds. Nie et al. [NWH14] imposed a new rank constraint to the Laplacian matrix, so that connected components in the resulting similarity matrix were exactly equal to the cluster number.

Multiple graph clustering has received increasing attention due to the availability of massive network data from diverse domains [LSJH18]. Early approaches either collapsed multiple graphs into a weighted single graph, or extended the existing algorithms for each graph and then merged partitions via consensus clustering. For example, Berlingerio et al [BCG11] counted the number of dimensions connecting any two nodes and weighted for the mono-dimensional edge. Papalexakis et al. [PAI13] integrated weighted

graphs based on a tensor factorization approach and used search heuristics to achieve the best clusters. Taylor et al. [TSSM16] applied a threshold to the entries of adjacency matrices to generate an unweighted adjacency matrix for layer aggregation. Tang et al. [TWL09] extracted structural features from each graph via modularity maximization and then integrated them by PCA to find community structures. However, these approaches have been criticized for ignoring the association features across different interactions, which are significant for improving the clustering performance. For example, quickly discovering drug-disease association can reduce the risk of drug discovery in traditional medicine and biology [ZYW*21].

2.3. Graph Visualization

Graph visualization has been extensively used to present pairwise relationships among objects. Nobre et al. [NMSL19] summarized three approaches for graph visualization: node-link layouts, tabular layouts, and implicit tree layouts. Node-link diagrams are the most common graphical representation of graphs. In a node-link diagram, nodes are drawn as point marks and links are drawn as line marks connecting the nodes [ZSS*21]. In a tabular layout, nodes and links are represented as columns and rows of a table [KLS*17]. The most well-known of this layout is the adjacency matrix [OJK19]. Implicit hierarchical layouts rely on node positioning to encode edges, and the well-known examples are TreeMaps [SSV18] and SunBursts [LWYXYYH13].

Graph clustering visualization methods are often designed to resolve the noisy natures of clustering results [YWL*21]. Most studies focused on the visual representation and comparison of clustering results. For example, DA-TU system [HHZ15] performed a force-directed placement on a clustered graph drawn in two dimensions. Lex et al. [LSS*12] compared different clusters using parallel coordinates for the exploration of large-scale heterogeneous genomics data. For multiple graphs, it is a challenge to represent and analyze the richness of information encoded in the multiple graphs compactly. Recent researches only focus on helping understand and interpret the multiple structures, while ignoring the associated features across multiple graphs. For example, Zhang et al. [ZWYY18] visualized the topological features within each layer network based on Louvain community detection. The hive plot [YTL*16] was introduced to present four layers of health data, in which edges are displayed between adjacent axes and vertices on the horizontal axis are colored based on their clusters. MuNeEye [HLZ*20] built a component view for summarizing all of the information obtained from the multi-layer-network calculation compactly.

3. Requirement Analysis and System Overview

In traditional multiple graph clustering methods, associations and varied local correlations of nodes across different graphs are usually ignored, consequently leading to poor clustering results. In this paper, we design a multiple graph clustering model to blend heterogeneous features and association information across multiple graphs holistically. In this section, the analysis requirements are listed and the overviewing of our visual clustering framework is presented.

3.1. Requirement Analysis

R1. The unified representation of multiple graphs. In traditional graph clustering strategies, single-graph features are merely considered or multi-graph features are simply weighted, which are insufficient for the construction of reasonable multiple graph clustering features, since multiple graphs may exhibit correlated or disparate clustering structures in each graph. It is difficult to construct a unified feature space, where multiple relationships of nodes can be comprehensively integrated and the varied local correlations within different graphs can be well balanced. How to quantify and combine structural characteristics of nodes across multiple graphs? How to balance the varied local correlations within different graphs?

R2. Multiple graph clustering with association features preserved. Traditional multiple graph clustering methods often focus on integrating heterogeneous features from multiple graphs to improve the accuracy of clustering results, but neglect the association features between pairwise graphs which are significant to obtain more reliable clustering features. It is also difficult to conduct multiple graph clustering to achieve reasonable clustering features with association features across multiple graphs preserved. How to capture association features between pairwise graphs? How to preserve association features in the clustering process?

R3. Interactive optimization of multiple graph clustering features. In specific application scenes, the importance of each single graph is not only determined through the clustering model, which would also be weighted according to the user requirements. For example, if the users are interested in the clustering features within a graph, they can highlight the features of interest in the clustering results, also with the features considered in the other graphs. Thus, it is quite necessary to enable users to sift through combinations of clustering parameter space and dynamically steer the clustering results to their analysis goals. How to estimate and adjust the importance of each graph? How to optimize the clustering features based on specific clustering objectives?

R4. Evaluation of multiple graph clustering features. Multiple graph clustering model aims to tackle a trade-off problem, which inevitably brings trouble for the evaluation of clustering results. For example, multiple relationships and association features in the clustering results cannot be well measured, especially for those complex multiple graphs. Thus, it is obviously difficult to evaluate and compare the validity of clustering results in multiple relationship integration and association feature preservation for multiple graphs. How to visually present the multiple relations in each graph and association features across multiple graphs? How to demonstrate clustering features obtained through our multiple graph clustering model outperform other models?

3.2. System Overview

Motivated by the identified requirements, we designed an interactive multiple graph clustering framework, enabling users to conduct clustering, evaluation and optimization on multiple graphs. The pipeline of our clustering system is presented in Figure 2. Firstly, multiple graph dataset is loaded into the clustering system. Then, node similarity matrix of each graph is constructed through graph

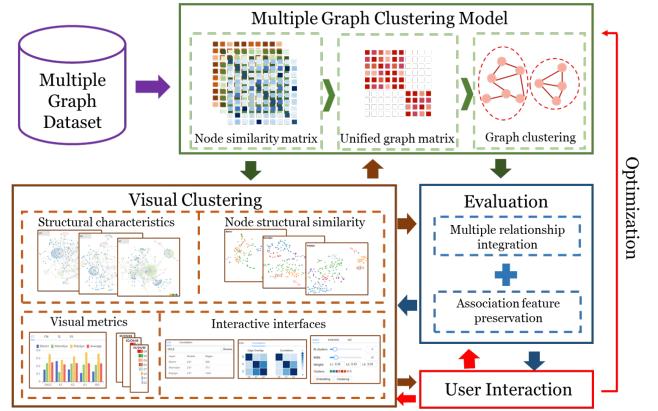


Figure 2: The pipeline of iMGC, which is comprised of multiple graph clustering, visual clustering and evaluation.

representation learning and further combined into a unified similarity matrix by balancing varied local correlations between different graphs (**R1**). Based on the unified similarity matrix, constrained Laplacian rank is applied to generate clustering results in which multiple relations and associations of nodes across multiple graphs are well preserved (**R2**). Furthermore, a group of quantitative and visual metrics are utilized to evaluate the performance of the proposed clustering strategy in multiple relationship integration and association feature preservation (**R4**). A set of visualization and interaction interfaces are provided for users to intuitively optimize and evaluate the multiple graph clustering features, and interactively explore the multiple graphs (**R3, R4**).

4. Multiple Graph Clustering Model

There are four components consisted in the proposed multiple graph clustering model, including structural characteristic estimation, unified graph matrix construction, multiple graph clustering and optimization. The architecture is presented in Figure 3. The main technologies and theories of each component are presented in this section.

4.1. Structural Characteristics Estimation

In recent years, graph representation learning (GRL) is widely applied to learn vectors to embed structural characteristics within the networks. As one of the state-of-art models for GRL, node2vec [GL16] preserves higher-order proximities by maximizing the probability of occurrence of subsequent nodes in fixed-length random walks. In this paper, for a multiple graph dataset with m graphs, we utilize node2vec to embed structural information of each graph into quantitative vectors. Then, the pairwise node similarity can be calculated according to the distance of corresponding pairwise vectors, through which the multiple relations of different graphs are expressed comprehensively. In the course of node2vec embedding, even disparate nodes have a certain degree of similarity between them, while those closely connected nodes would also contain noisy connectivity, such as irrelevant or spurious connectivity, which will affect the effectiveness of clustering features. Con-

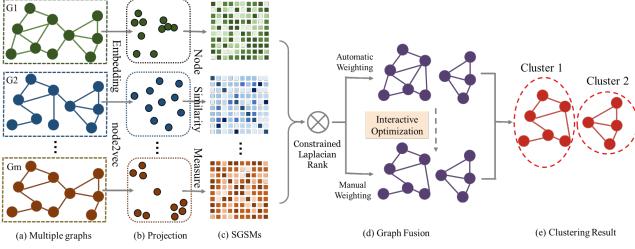


Figure 3: Illustration of our multiple graph clustering model. Given a multiple graph data with m graphs (a), we first capture and quantify structural characteristics by node2vec (b) and node similarity matrix of each graph is achieved by the distance of corresponding vectors (c). Two schemes are provided for graph fusion to generate the unified graph matrix (d), including automatic and manual weighting. The final clusters are provided directly on the unified graph matrix by constrained Laplacian rank optimization (e).

sidering that sparse representation is robust to noise and outliers, we introduce an adaptive k -nearest neighbor algorithm [SH10] to construct the single-graph similarity matrices (SGSMs), so that each node connects with k nodes. It is desired to construct a SGSM of each graph that a smaller distance between two nodes corresponds to a large similarity value, and a large distance between two nodes corresponds to a small similarity value.

4.2. Unified Graph Matrix Construction

Based on the SGSMs, we introduce a graph fusion method [WYL19]. Each graph can be weighted automatically and the unified graph matrix is learned jointly so that they can help each other in a mutual reinforcement manner. To balance the varied local correlations within different graphs, the unified graph matrix is constructed by minimizing the Frobenius norm of the difference between the unified graph matrix and the single-graph node similarity matrix.

$$\begin{aligned} \min_{\mathbf{U}} \sum_{v=1}^m w_v \|\mathbf{U} - \mathbf{S}^v\|_F^2 \\ s.t. \forall i, u_{ij} \geq 0, \mathbf{1}^T \mathbf{u}_i = 1 \end{aligned} \quad (1)$$

where \mathbf{S}^v is the v_{th} graph of the SGSMs, u_{ij} is the j_{th} element of a column vector \mathbf{u}_i of the unified graph matrix \mathbf{U} , $\mathbf{1}$ denotes a column vector with all entries of one, w_v is the weight of the v_{th} SGSM S^v . The initial input values of the weights are all defined as $1/m$. We also provide an interactive weight parameter input interface for multiple graph optimization, which supports users to manually assign initial weights of multiple graphs whose sum is 1. When the users specify the initial weight values, the clustering results will be generated more inclined to user preferences, but the final weights are still determined by our multiple graph clustering model.

4.3. Multiple Graph Clustering

Given the unified graph matrix \mathbf{U} obtained by graph fusion, we produce the multiple graph clustering results directly on the Lapla-

cian matrix of the unified graph matrix by means of a rank constraint. In graph theory, $\mathbf{L}_U = \mathbf{D}_U - (\mathbf{U}^T + \mathbf{U})/2$ is called as the Laplacian matrix of the unified graph matrix, and the degree matrix $\mathbf{D}_U \in \mathbb{R}^{n \times n}$ is defined as a diagonal matrix where the i_{th} diagonal element is $\sum_j (U_{ij} + U_{ji})/2$. If \mathbf{U} is nonnegative, the Laplacian matrix \mathbf{L}_U will have an important property [MAC*91] as follows: *The multiplicity c of the eigenvalue 0 of the Laplacian matrix \mathbf{L}_U is equal to the number of connected components in the graph with the unified graph matrix \mathbf{U} .* Motivated by the property, we add an additional constraint $\text{rank}(\mathbf{L}_U) = n - c$, into the Equation 1 to achieve the desired node aggregation with the clear clustering structures.

$$\begin{aligned} \min_{\mathbf{U}} \sum_{v=1}^m w_v \|\mathbf{U} - \mathbf{S}^v\|_F^2 \\ s.t. \forall i, u_{ij} \geq 0, \mathbf{1}^T \mathbf{u}_i = 1, \text{rank}(\mathbf{L}_U) = n - c \end{aligned} \quad (2)$$

However, the constraint $\text{rank}(\mathbf{L}_U) = n - c$ is difficult to tackle, because the Laplacian matrix $\mathbf{L}_U = \mathbf{D}_U - (\mathbf{U}^T + \mathbf{U})/2$ and the degree matrix \mathbf{D}_U both depend on the unified graph matrix \mathbf{U} . According to the Ky Fan's Theorem [FK50], the Equation 2 can be transformed to an optimization problem to minimize the weights of cut edges, which is much easier to solve:

$$\begin{aligned} \min_{\mathbf{U}} \sum_{v=1}^m w_v \|\mathbf{U} - \mathbf{S}^v\|_F^2 + 2\lambda \text{Tr}(\mathbf{F}^T \mathbf{L}_U \mathbf{F}) \\ s.t. \forall i, u_{ij} \geq 0, \mathbf{1}^T \mathbf{u}_i = 1, \text{rank}(\mathbf{L}_U) = n - c, \mathbf{F}^T \mathbf{F} = 1 \end{aligned} \quad (3)$$

where λ is the parameter, \mathbf{F} is the embedding matrix and $\text{Tr}()$ denotes the trace of the matrix.

4.4. Optimization

Traditionally, one variable is optimized via the Augment Lagrange Multiplier (ALM) scheme [BD82], under the assumption that other variables have been obtained. However, it is difficult to solve the Equation 3, because all the variables in the objective function are coupled together. Therefore, we introduce an alternative iterative algorithm to solve the problem. In practice, we first initialize all variations, and then iteratively update the following three problems in turn until the rank constraint is satisfied or the maximum number of iterations is reached. Finally, the learned unified graph matrix \mathbf{U} is obtained with c connected components that are the final clusters. Specifically, take w as an example. When \mathbf{U} and \mathbf{F} are fixed, Equation 3 for updating w is to solve the Equation 1, then take the derivative of the Lagrange function of Equation 1 with respect of \mathbf{U} and set the derivative to zero, so that

$$w_v = \frac{1}{2\sqrt{\|\mathbf{u} - \mathbf{s}^v\|_F^2}} \quad (4)$$

As we can see, the weight of each graph w_v is affected by the degree of association between pairwise similarity matrices, and they are learned jointly in a reinforcement manner. For example, if the paired graphs are very similar, the weights of the two graphs will

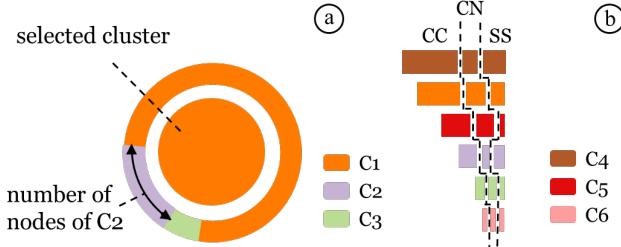


Figure 4: Visual designs for the cluster quality. The a and b present the glyph-based visualization and the stacked bar chart-based visualization respectively.

increase accordingly, so that the varied local correlations between different graphs are well balanced. Similarly, fix w and \mathbf{F} , update \mathbf{U} by solving the following Equation:

$$\begin{aligned} \min_{\mathbf{u}_i} \sum_{v=1}^m & \left\| \mathbf{u}_i - \mathbf{s}_i^v + \frac{\lambda}{2mw_v} \mathbf{d}_i \right\|_2^2 \\ \text{s.t. } & \forall i, u_{ij} \geq 0, \mathbf{1}^T \mathbf{u}_i = 1 \end{aligned} \quad (5)$$

Fix w and \mathbf{U} , update \mathbf{F} which is formed by the c eigenvectors of \mathbf{L}_U corresponding to the c smallest eigenvalues. It can be seen that our final clusters are produced in the unified graph matrix with no additional clustering steps, which preserves the associated structures across multiple graphs to the greatest extent.

5. Visual Interfaces

To meet the design requirements discussed in Section 3, we develop iMGC, providing a rich set of visual designs and interactions for the visual evaluation (**R4**) and optimization (**R3**) of multiple graph clustering results.

5.1. Visual Designs

The visual cues of iMGC are designed to provide an overview of multiple graphs and their clustering features in detail, enabling users to visually evaluate clustering results from a variety of perspectives (**R4**).

Graph Clustering Model. We compare our clustering method (iMGC) with the single graph clustering method (KMEANS) and the weighted summing clustering method (WS). For different clustering models, different combinations of parameter space are provided in the control panel. For instance, our method (iMGC) provides three parameters: the number of clusters (N-clusters), the initial value of K in the adaptive k -nearest neighbor algorithm (k -NN), and the initial weight value of each graph (weight). For k -means, two parameters are provided: N-clusters and the graph with k -means applied (layer). WS provides only one parameter: N-clusters.

Graph Clustering Visualization. Node-link diagrams, which present the natural structural characteristics of multiple graphs, are employed to layout multiple graphs with a force-directed model, as shown in Figure 1. To visually present the node similarity of each

graph in the vectorized space, we utilize scatterplots to visualize the projections of vectors based on a dimensionality reduction method t-SNE, where geometric distance of pairwise points presents the structural similarity between nodes in the node-link diagram. When a clustering scheme is specified, nodes in node-link diagrams and scatterplots are shaded in different colors according to their clusters. Therefore, the effectiveness of the integration with multiple relationships and balance of varied local correlations across multiple graphs can be visually evaluated based on the visual distribution of nodes in the same color in the multiple node-link diagrams and projection views.

Metrics Visual Presentation. We also design a set of visual cues to present clustering metrics and how they are affected by weight changes and different clustering strategies. First, the evaluation view presents the visual quality comparison of different graphs under different clustering methods or weight assignments in the form of histograms. Second, as shown in Figure 4a, the glyph is composed of two parts: the inner circle color depicts the selected cluster and the outer ring represents the proportion of different cluster nodes surrounded by the silhouette of nodes in the selected cluster on the projection view, where the color of arc encodes different clusters, and the length encodes the number of nodes. The more uniform the color of the outer ring and the inner circle, the better the correlation of nodes in a cluster is retained in the clustering results. Third, as shown in Figure 4b, the stacked bar charts visually present the total quality metrics of each cluster, including Clustering Coefficient (CC), Silhouette Coefficient (SC) and Connected Component (CN). The height of each bar encodes the total value, and the colors correspond to different clusters.

5.2. Interaction

The iMGC also supports the following user interactions to help users optimize clustering results (**R3**). First, we develop a set of interactions to help users switch between the coordinated views. For example, in the data view, users can switch data information view (Info) and correlation view (Correlation view) in which the edge overlap [dNAL15] and average degree-degree correlation [CS10] between pairwise graphs are presented in the form of the heatmap. In the control panel, users can change the clustering method by clicking the desired method in the drop-down box. In addition, we develop a focus mode enabling users to obtain an intuitive perspective of association features across multiple graphs. When users click a rectangle arranged in a row in the control panel, not only will the glyph of the selected cluster be appended to the projection view, but all the other clusters will be colored as light grey to highlight the selected cluster. In the evaluation view, users can compare the clustering quality of multiple graphs in the perspectives of different metrics (DF) or different weight assignments (DW) by switching between DF and DW. Based on the knowledge of the multiple graph dataset and visual evaluation of adaptive clustering results, users can adjust clustering features according to their preferences. The control panel provides several configurable parameters for users to adjust and optimize the clustering features by entering numbers in the input box including the number of clusters and the initial weight of each graph.

Table 1: Quantitative comparison of different clustering schemes based on multiple graph datasets

Quality Metrics			CC				Q				SC				CN			
Dataset	Cluster Number	Method	G1	G2	G3	Ave	G1	G2	G3	Ave	G1	G2	G3	Ave	G1	G2	G3	Ave
MOSS	5	iMGC	0.015	0.011	0.010	0.012	0.351	0.419	0.456	0.409	-0.225	0.292	0.313	0.127	61	20	25	35
		<i>k</i> -means1	0.021	0.006	0.005	0.011	0.645	0.238	0.249	0.378	0.272	-0.052	-0.096	0.041	5	171	184	120
		<i>k</i> -means2	0.000	0.004	0.008	0.004	0.221	0.571	0.329	0.374	-0.141	0.025	0.052	-0.021	174	6	131	103
		<i>k</i> -means3	0.004	0.008	0.005	0.006	0.204	0.252	0.389	0.282	-0.141	0.025	0.052	-0.021	191	158	111	153
		WS	0.014	0.007	0.004	0.008	0.417	0.282	0.280	0.326	-0.035	-0.101	-0.168	-0.101	108	144	173	141
	6	iMGC	0.015	0.011	0.010	0.012	0.337	0.400	0.457	0.398	-0.231	0.265	0.310	0.115	67	29	23	39
		<i>k</i> -means1	0.021	0.007	0.001	0.010	0.663	0.235	0.237	0.379	0.386	-0.063	-0.103	0.073	6	180	193	126
		<i>k</i> -means2	0.000	0.005	0.005	0.004	0.228	0.579	0.333	0.380	-0.114	0.266	-0.096	0.019	182	6	134	107
		<i>k</i> -means3	0.004	0.008	0.010	0.007	0.158	0.194	0.306	0.219	-0.144	0.008	0.016	-0.040	240	206	169	205
	7	WS	0.014	0.003	0.009	0.009	0.342	0.225	0.234	0.267	-0.075	-0.152	-0.215	-0.147	161	204	208	191
		iMGC	0.022	0.016	0.010	0.016	0.180	0.318	0.357	0.285	-0.237	0.168	0.309	0.080	75	21	28	41
		<i>k</i> -means1	0.016	0.009	0.000	0.008	0.672	0.230	0.221	0.374	0.398	-0.081	-0.129	0.063	7	193	205	135
		<i>k</i> -means2	0.000	0.005	0.005	0.004	0.183	0.530	0.279	0.331	-0.124	0.153	-0.118	-0.030	223	47	180	150
		<i>k</i> -means3	0.007	0.005	0.004	0.005	0.152	0.192	0.317	0.220	-0.173	-0.030	0.011	-0.064	252	217	172	213
		WS	0.015	0.003	0.009	0.009	0.344	0.235	0.240	0.273	-0.088	-0.171	-0.238	-0.166	169	207	213	196
CELE	8	iMGC	0.155	0.238	0.355	0.250	0.281	0.406	0.333	0.340	-0.216	0.012	0.122	-0.027	62	15	14	30
		<i>k</i> -means1	0.266	0.123	0.280	0.223	0.621	0.163	0.137	0.307	0.005	-0.108	-0.089	-0.064	10	78	42	43
		<i>k</i> -means2	0.158	0.234	0.326	0.239	0.288	0.472	0.287	0.349	-0.153	0.055	0.014	-0.028	83	9	21	37
		<i>k</i> -means3	0.112	0.220	0.377	0.236	0.266	0.374	0.364	0.335	-0.141	-0.040	0.177	-0.001	80	27	8	38
		WS	0.131	0.217	0.362	0.237	0.342	0.379	0.300	0.340	-0.147	-0.015	0.056	-0.035	48	29	20	32
	9	iMGC	0.172	0.242	0.371	0.262	0.307	0.401	0.323	0.344	-0.182	0.001	0.085	-0.032	71	20	20	36
		<i>k</i> -means1	0.282	0.073	0.264	0.206	0.627	0.126	0.121	0.291	0.056	-0.142	-0.128	-0.072	13	98	62	57
		<i>k</i> -means2	0.141	0.253	0.336	0.243	0.252	0.469	0.273	0.331	-0.177	0.064	0.009	-0.034	94	12	26	44
		<i>k</i> -means3	0.107	0.219	0.394	0.240	0.266	0.354	0.360	0.327	-0.145	-0.059	0.135	-0.023	84	31	10	41
	10	WS	0.134	0.219	0.368	0.240	0.346	0.376	0.292	0.338	-0.145	-0.057	0.015	-0.062	51	30	21	34
		iMGC	0.153	0.245	0.381	0.260	0.302	0.405	0.329	0.345	-0.191	0.000	0.079	-0.037	73	20	20	37
		<i>k</i> -means1	0.264	0.093	0.260	0.206	0.612	0.131	0.118	0.287	0.024	-0.152	-0.140	-0.089	13	102	59	58
		<i>k</i> -means2	0.122	0.242	0.330	0.231	0.225	0.464	0.261	0.317	-0.193	0.028	-0.023	-0.063	102	12	29	47
		<i>k</i> -means3	0.101	0.199	0.389	0.230	0.247	0.345	0.345	0.312	-0.153	-0.098	0.078	-0.058	93	35	10	46
		WS	0.135	0.224	0.341	0.233	0.338	0.366	0.287	0.330	-0.163	-0.058	-0.005	-0.076	59	34	22	38

6. Evaluation

In this section, two multiple graph datasets are experimented to evaluate the effectiveness of our clustering method. Then, case studies are conducted based on three real-word datasets to verify the validity of our visual clustering system.

6.1. Datasets

Three real-world datasets (D1, D2, and D3) are utilized to demonstrate the effectiveness of our multiple graph clustering method. D1 (CELE) is a constructed from *caenorhabditis elegans* connectome dataset, which consists of 237 nodes, forming three networks based on 1361 edges through different synaptic junctions: electric (ElectrJ), chemical monadic (MonoSyn) and polyadic (PolySyn). D2 (MOSS) is a twitter network focused on the 2013 World Championships in Athletics, which contains 400 vertices and 518 edges. we form a three-layer graph from three different types of social relationships among users including retweet (RT), mentions (MT) and replies (RE). D3 is a bibliographic dataset including up to 828 vertices and 5175 edges. A vertex stands for a paper, and an edge is formed if two papers are written by the same authors or have citation relationships.

6.2. Quantitative Comparison

We have conducted a set of experiments for comparing three different graph clustering strategies, including our clustering method (iMCG), single graph clustering method (*k*-means) [HW79] and the weighted summing clustering method (WS) [CWY18]. It should

be noted that *k*-means1 refers to that clustering features obtained by *k*-means on the first graph directly act on other graphs, and *k*-means2 and *k*-means3 are similar. Four metrics, Clustering Coefficient (CC) [Döh19], Modularity (Q) [NG04], Silhouette Coefficient (SC) [Rou87] and Connected Component (CN), are used to measure multiple graph clustering performance. CN sums the number of connected components in all clusters, so that the higher the value, the more closely the nodes in the cluster are connected. To assess multiple graph clustering quality, we compute the indicator scores for each graph (G1, G2, G3) and average scores (AVE).

Table 1 presents the four metrics on G1, G2, G3, and AVE obtained by three clustering strategies for the datasets of MOSS and CELE. The number of clusters is specified as 5 to 7 on MOSS and 8 to 10 on CELE. (1) *Average performance* (AVE). Obviously, our method performs better on all metrics, especially on CC and CN. In particular, for the CELE in which the original structures present sparse connections, if the number of clusters is small, *k*-means2 and *k*-means3 perform better on the metrics of Q and SC. However, when the cluster number increases, our method regains its superiority. It demonstrates that our clustering method effectively integrates structural characteristics and balances the varied local correlations of multiple graphs. (2) *Performance on each graph* (G1, G2, G3). First, *k*-means usually performs better than our method on the corresponding single graph, but performs poorly on other graphs. It is surprising that our method often outperforms *k*-means2 and *k*-means3 on the G2 and G3 because of the high degree of association of G2 and G3. Second, as can be observed, the performance of WS is acceptable, improving the clustering quality in multiple graphs.

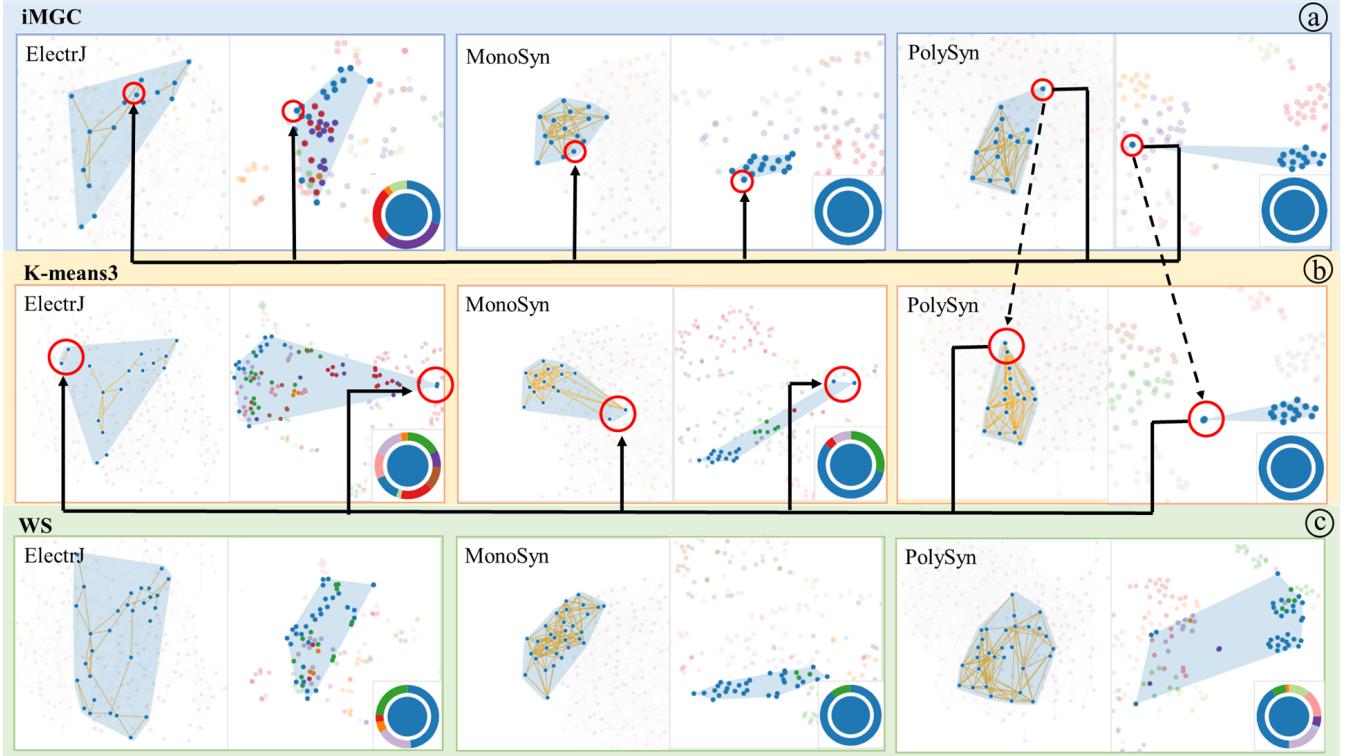


Figure 5: Comparison of the clustering quality of similar discovered clusters obtained by different methods in CELE dataset, including iMGC (a), k-means3 (b), and WS (c).

Only in G1, WS outperforms our method on the metric of Q because of low association. However, WS underperforms our method in most cases. Only in G1, WS outperforms our method on the metric of Q in G1 because of the low association between G1 and the other graphs. The above results demonstrate that our clustering method effectively preserves the association features across multiple graphs.

6.3. Case Study

Graph clustering plays significant roles in the community detection in social networks and functional module identification in biological graphs. In our case studies, we use the aforementioned two social graphs and one biological graph to validate the effectiveness of the proposed multiple graph clustering method.

Case 1: Multiple Graph Clustering

To investigate whether the proposed graph clustering model is able to achieve reasonable clusters with the multiple relationships of nodes and associations across multiple graphs taken into consideration, we conducted a detailed analysis on the CELE dataset through comparisons with the other methods on the overall performance and detailed presentation of a discovered cluster.

First, we perform initial clustering with 8 clusters (users can specify any clustering number, and here we chose an arbitrary number). From the node-link diagrams and projection views, we acquire an overall understanding of clustering features. As shown in

Figure 1, the nodes of the same cluster obtained by iMGC are connected more densely and projected more closely, compared with *k*-means1 and WS. Similar conclusions can also be drawn from the Evaluation View (Figure 5e). Furthermore, the detailed features of a discovered cluster with the best indicators in MonoSyn are shown in Figure 5. In detail, there are two nodes of the blue cluster obtained by *k*-means3, which are different from our results, as shown in the red circles highlighted in Figures 5(a, b). Two nodes appear to be more densely connected to the cluster in PolySyn, but they are quite irrelevant in the ElectrJ and MonoSyn. Then, as shown in Figure 5c, although WS can improve the connectivity of nodes of a cluster on the node-link diagram, the nodes are very scattered on the projection view. By contrast, the nodes of blue cluster obtained by iMGC are densely connected and closely projected which are highlighted with polygons. The experimental results indicate that our method performs better in the balance of varied structural characteristics within different graphs and the preservation of association features across multiple graphs.

Case2: Interactive Optimization

The weight of each graph has a great impact on the clustering results of multiple graphs. Thus, we compare the clustering performance before and after adjusting weights to prove the effectiveness of our interactive optimization system.

MOSS dataset is a network connecting users on the basis of the retweets (RT), mentions (MT) and replies (RE). We first generate clustering results by adaptively learning weights, where the

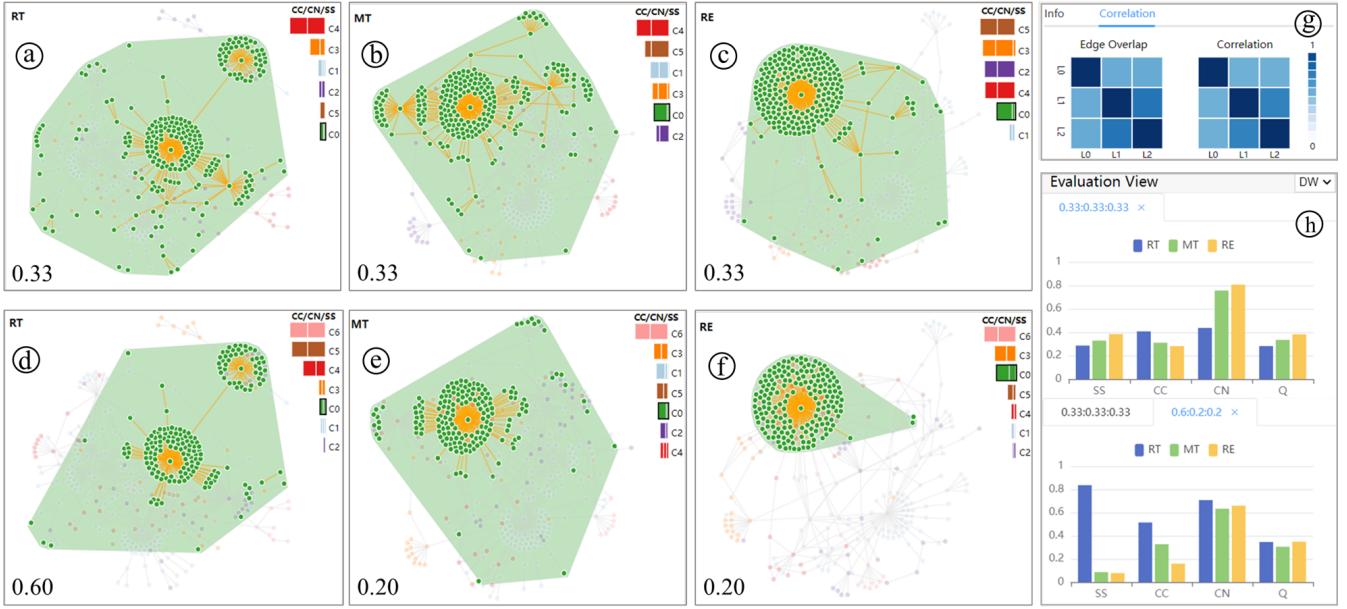


Figure 6: An example of interactive weight adjustment for the optimization of multiple graph clustering. (a-c) present a cluster of interest aggregated by a group of default initial weights (RT:MT:RE=0.33:0.33:0.33). (d-f) present a cluster of interest obtained by a group of adjusted weights (RT:MT:RE=0.6:0.2:0.2). (g) and (h) respectively present the data overview of the MOSS dataset and the clustering quality under different weights in the form of histograms.

initial weight of each graph is one third. Obviously, as shown in Figure 6, the clustering structure in RT does not perform as well as MT and RE. There are many isolated scattered nodes in the green cluster (Figure 6a), because of the higher degree of association between MT and RE, which can be easily explained by the heatmap (Figure 6g). Although a highly retweeted user is likely not to be mentioned or replied to by so many users, the behavior of retweet express stronger emotion, and it can better reflect the closeness between people in an event. Therefore, the adaptive results that only consider the association information between graphs while ignoring the background knowledge of data do not seem to be in line with the actual situation. It is necessary to increase the initial weight of the RT. The initial weight assignment is adjusted as RT:MT:RE=0.6:0.2:0.2. As shown in Figures 6(d-f), the star topology structures in all graphs are completely preserved, and the number of isolated nodes in RT is also reduced, which is in line with our expectations and real-world data background knowledge. The above interactive optimization operations and results show that adjusting the initial weights of multiple graphs according to the relevant knowledge can enhance or weaken the importance of a graph in the process of clustering, so as to meet the clustering requirements in a specific context.

Case3: Multiple Graph Exploration

It is an interesting problem to exploit the multiple sources of information to make better inferences about entities and their relationships. In our experiments, we will get more comprehensive and reliable insights by integrating information from multiple graphs including citation network (Citation) and co-author network (Co-Author). As shown in Figure 7c, in the Citation network, it is

strange that there is an isolated node in the yellow cluster which indicates that there are no citation relationships between the isolated node and the other nodes in the cluster. However, from the co-author network, it can be seen that the isolated node is closely connected with the cluster. In order to explain this finding, we further explore the details of the node and cluster. Specifically, by extracting the keywords of the paper titles in the cluster, as shown in Figure 7c, it is found that they are mostly likely to belong to computer graphics. And the title of the isolated paper is “Physically-based methods for polygonization of implicit surfaces”, which really belongs to the computer graphics, that is, it matches the feature of the yellow cluster, as shown in Figure 7d. Another similar situation is shown in Figures 7(e, f). Although a paper has no citation relationship with the red cluster composed of papers in the field of the computer system, our algorithm further learns the co-author relationship and successfully aggregates it into the red cluster. Therefore, it can be proved that our method can effectively integrate multiple information and help to obtain more reasonable and credible insights.

6.4. Discussion

Compared with traditional clustering strategies, the proposed multiple graph clustering model performs better on most of the metrics. The main advantages are derived from that we conduct a constrained Laplacian rank optimization on the process of multiple graph clustering, through which the association information between pairwise graphs are well learned. In addition, an interactive optimization mechanism is designed for users to adjust the importance ratio of different graphs based on the dataset background knowledge for obtaining more reasonable and credible in-

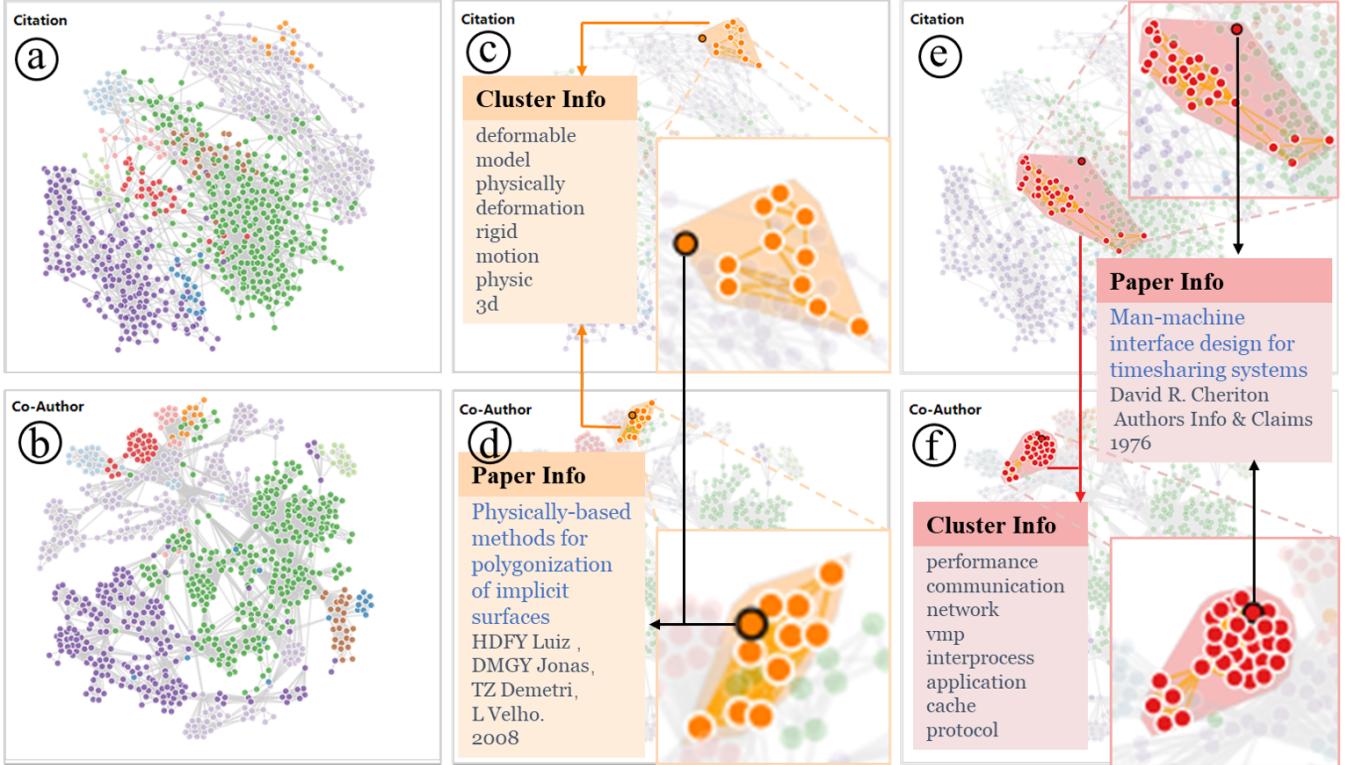


Figure 7: Clustering feature exploration. a and b are the overall clustering results in a bibliographic dataset. c-d and e-f show the specific information of the orange and red clusters respectively.

sights from the multiple graph clustering results. Therefore, our clustering method cannot only leverage the association information among multiple graphs, but also further optimize the clustering results according to the actual situation. There are still some issues not well resolved as follows: (1) In this paper, multiple graph datasets are all regular, with the common nodes and different topological relationships. However, multiple graphs might involve complex representations in real scenarios. For example, nodes are different and changed dynamically. Therefore, how to further expand the multiple graph clustering algorithm for complex scenarios is a topic worthy of study. (2) Multiple graph clustering is a trade-off problem, the results of which are consistent on the one graph, but might be contradictory on the others. Although a constrained Laplacian rank problem is optimized in this paper for fusing multiple graphs, there are still some inevitable errors occurring in the clustering results, which can be found from the results of quantitative comparison in section 6.2. Therefore, it is necessary to further study the clustering model which can better integrate structures and lose fewer features as far as possible in the clustering results. (3) In this paper, iMGC enables users to adjust the parameters interactively according to their requirements and then realize tendentious clustering results. However, the user interaction is quite dependent on the prior knowledge of the underlying data, which would not be mastered by common users. Therefore, it is necessary to explore the mapping relationship between users' goal and parameters and establish a mechanism for users to specify the goal and adaptively

adjust parameters, so as to assist users to quickly obtain their desired multiple graph clustering results.

7. Conclusion

In this paper, a unified graph matrix is constructed to combine the structural characteristics of each graph and a constrained Laplacian rank is utilized to generate the multigraph clustering features from the unified graph directly, in which the associations between pairwise graphs are well preserved. Also, a group of metrics are employed to measure the validity of our clustering model in multiple relationship integration and association feature preservation. In addition, a set of visualization and interaction interfaces are provided to enable users to intuitively optimize and evaluate the multiple graph clustering features, and interactively explore the multiple graphs. Case studies and quantitative comparisons based on real-world datasets have demonstrated the effectiveness of iMGC in the clustering performance from various perspectives and exploration of multiple graphs. In the future work, we will expand the multiple graph clustering algorithm for complex scenarios, study the clustering model which can better integrate structures and lose fewer features, and establish a mechanism for users to specify the goal and adaptively adjust parameters.

References

- [BCG11] BERLINGERIO M., COSCIA M., GIANNOTTI F.: Finding and characterizing communities in multidimensional networks. In *2011 In-*

- ternational Conference on Advances in Social Networks Analysis and Mining (2011), pp. 490–494. doi:[10.1109/ASONAM.2011.104.3](https://doi.org/10.1109/ASONAM.2011.104.3)
- [BD82] BERTSEKAS, DIMITRI P.: *Constrained optimization and Lagrange multiplier methods*. Constrained optimization and Lagrange multiplier methods, 1982. 5
- [BN02] BELKIN M., NIYOGI P.: Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in Neural Information Processing Systems* (2002), Dietterich T., Becker S., Ghahramani Z., (Eds.), vol. 14, MIT Press. URL: <https://proceedings.neurips.cc/paper/2001/file/f106b7f99d2cb30c3db1c3cc0fde9ccb-Paper.pdf>. 3
- [BSZG18] BOJCHEVSKI A., SHCHUR O., ZUEGNER D., GÜNNEMANN S.: Netgan: Generating graphs via random walks. 1
- [CGL*19] CAI Y., GE L., LIU J., CAI J., CHAM T.-J., YUAN J., THALMANN N. M.: Exploiting spatial-temporal relationships for 3d pose estimation via graph convolutional networks. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (2019), pp. 2272–2281. doi:[10.1109/ICCV.2019.00236.1](https://doi.org/10.1109/ICCV.2019.00236.1)
- [CS10] C. SPEARMAN: The proof and measurement of association between two things. *International Journal of Epidemiology* (2010). 6
- [CWY18] CHAI D., WANG L., YANG Q.: Bike flow prediction with multi-graph convolutional networks. In *Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems* (New York, NY, USA, 2018), SIGSPATIAL ’18, Association for Computing Machinery, p. 397–400. URL: <https://doi.org/10.1145/3274895.3274896>, doi:[10.1145/3274895.3274896.7](https://doi.org/10.1145/3274895.3274896.7)
- [dNAL15] DE DOMENICO M., NICOSIA V., ARENAS A., LATORA V.: Structural reducibility of multilayer networks. *Nature Communications* 6 (Apr. 2015), 6864. doi:[10.1038/ncomms7864.6](https://doi.org/10.1038/ncomms7864.6)
- [Döh19] DÖHNE M.: *Luce/Perry (1949): A Method of Matrix Analysis of Group Structure*. Springer Fachmedien Wiesbaden, Wiesbaden, 2019, pp. 363–365. URL: https://doi.org/10.1007/978-3-658-21742-6_83, doi:[10.1007/978-3-658-21742-6_83.7](https://doi.org/10.1007/978-3-658-21742-6_83.7)
- [FK50] FAN, K.: On a theorem of weyl concerning eigenvalues of linear transformations ii. *Proceedings of the National Academy of Sciences* 35, 1 (1950), 652–655. 5
- [GCXW20] GONG M., CHEN C., XIE Y., WANG S.: Community preserving network embedding based on memetic algorithm. *IEEE Transactions on Emerging Topics in Computational Intelligence* 4, 2 (2020), 108–118. doi:[10.1109/TETCI.2018.2866239.3](https://doi.org/10.1109/TETCI.2018.2866239.3)
- [GL16] GROVER A., LESKOVEC J.: Node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2016), KDD ’16, Association for Computing Machinery, p. 855–864. URL: <https://doi.org/10.1145/2939672.2939754>, doi:[10.1145/2939672.2939754.3,4](https://doi.org/10.1145/2939672.2939754.3,4)
- [HZH15] HE S., HUANG M. L., ZHU L.: Natural user interface design in da-tu: An interactive clustered data visualization system. In *2015 19th International Conference on Information Visualisation* (2015), pp. 83–88. doi:[10.1109/IV.2015.25.3](https://doi.org/10.1109/IV.2015.25.3)
- [HLZ*20] HAN D., LU D., ZHENG S., JIANG H., WU Y.: Latent patterns detection and interpretation in multi-layer temporal network. *IEEE Access* 8 (2020), 132786–132798. doi:[10.1109/ACCESS.2020.3010693.3](https://doi.org/10.1109/ACCESS.2020.3010693.3)
- [HW79] HARTIGAN J., WONG M.: *A K-means Clustering Algorithm: Algorithm AS 136*, vol. 28. 01 1979, pp. 100–108. doi:[10.2307/2346830.3,7](https://doi.org/10.2307/2346830.3,7)
- [HYZ*19] HAN P., YANG P., ZHAO P., SHANG S., LIU Y., ZHOU J., GAO X., KALNIS P.: Gen-mf: Disease-gene association identification by graph convolutional networks and matrix factorization. pp. 705–713. doi:[10.1145/3292500.3330912.1](https://doi.org/10.1145/3292500.3330912.1)
- [KLS*17] KERZNER E., LEX A., SIGULINSKY C. L., URNESS T., JONES B. W., MARC R. E., MEYER M. D.: Graffinity: Visualizing connectivity in large graphs. *CoRR abs/1703.07729* (2017). URL: <http://arxiv.org/abs/1703.07729>, arXiv:1703.07729. 3
- [KRUT09] KUBY M., ROBERTS T., UPCHURCH C., TIERNEY S.: Network analysis. In *International Encyclopedia of Human Geography*, Kitchin R., Thrift N., (Eds.). Elsevier, Oxford, 2009, pp. 391–398. doi:<https://doi.org/10.1016/B978-008044910-4.00481-8.3>
- [LG19] LI C., GOLDWASSER D.: Encoding social information with graph convolutional networks for political perspective detection in news media. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (Florence, Italy, July 2019), Association for Computational Linguistics, pp. 2594–2604. URL: <https://aclanthology.org/P19-1247>, doi:[10.18653/v1/P19-1247.1](https://doi.org/10.18653/v1/P19-1247.1)
- [LSJH18] LIU W., SUZUMURA T., JI H., HU G.: Finding overlapping communities in multilayer networks. *PloS one* 13, 4 (2018), e0188747. URL: <https://europepmc.org/articles/PMC5919045>, doi:[10.1371/journal.pone.0188747.3](https://doi.org/10.1371/journal.pone.0188747.3)
- [LSS*12] LEX A., STREIT M., SCHULZ H.-J., PARTL C., SCHMALSTIEG D., PARK P., GEHLENBOORG N.: Stratome: Visual analysis of large-scale heterogeneous genomics data for cancer subtype characterization. *Computer Graphics Forum (EuroVis)* 31, 3 (2012), 1175–1184. doi:[10.1111/j.1467-8659.2012.03110.x.3](https://doi.org/10.1111/j.1467-8659.2012.03110.x.3)
- [LWYXYH13] LI-WEI G., YI C., XIN-YUE Z., YUE-HONG S.: A hierarchical data visualization algorithm: Self-adapting sunburst algorithm. In *2013 International Conference on Virtual Reality and Visualization* (2013), pp. 185–190. doi:[10.1109/ICVRV.2013.36.3](https://doi.org/10.1109/ICVRV.2013.36.3)
- [MAC*91] MOHAR B., ALAVI Y., CHARTRAND G., OELLERMANN O., SCHWENK A.: The laplacian spectrum of graphs. *Graph Theory, Combinatorics and Applications* 2 (01 1991), 5364. 5
- [NG04] NEWMAN M. E. J., GIRVAN M.: Finding and evaluating community structure in networks. *Phys. Rev. E* 69 (Feb 2004), 026113. URL: <https://link.aps.org/doi/10.1103/PhysRevE.69.026113>, doi:[10.1103/PhysRevE.69.026113.7](https://doi.org/10.1103/PhysRevE.69.026113.7)
- [NMSL19] NOBRE C., MEYER M., STREIT M., LEX A.: The state of the art in visualizing multivariate networks. *Computer Graphics Forum* 38 (06 2019), 807–832. doi:[10.1111/cgf.13728.3](https://doi.org/10.1111/cgf.13728.3)
- [NWH14] NIE F., WANG X., HUANG H.: Clustering and projected clustering with adaptive neighbors. KDD ’14, Association for Computing Machinery, p. 977–986. URL: <https://doi.org/10.1145/2623330.2623726>, doi:[10.1145/2623330.2623726.3](https://doi.org/10.1145/2623330.2623726.3)
- [OIO*19] OYELADE J., ISEWON I., OLADIPUPO O., EMEBO O., OMOBADEGUN Z., AROMOLARAN O., UWOGHIREN E., OLANIYAN D., OLAWOLE O.: Data clustering: Algorithms and its applications. In *2019 19th International Conference on Computational Science and Its Applications (ICCSA)* (2019), pp. 71–81. doi:[10.1109/ICCSA.2019.00001-3](https://doi.org/10.1109/ICCSA.2019.00001-3)
- [OJK19] OKOE M., JIANU R., KOBOUROV S.: Node-link or adjacency matrices: Old question, new insights. *IEEE Transactions on Visualization and Computer Graphics* 25, 10 (2019), 2940–2952. doi:[10.1109/TVCG.2018.2865940.3](https://doi.org/10.1109/TVCG.2018.2865940.3)
- [PAI13] PAPALEXAKIS E. E., AKOGLU L., IENCE D.: Do more views of a graph help? community detection and clustering in multi-graphs. In *Proceedings of the 16th International Conference on Information Fusion* (2013), pp. 899–905. 3
- [PAS14] PEROZZI B., AL-RFOU R., SKIENA S.: Deepwalk: Online learning of social representations. *CoRR abs/1403.6652* (2014). URL: <http://arxiv.org/abs/1403.6652>, arXiv:1403.6652. 3
- [RG02] RICHARDSON S., GREEN P.: On bayesian analysis of mixtures with an unknown number of components. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 60 (01 2002), 661 – 661. doi:[10.1111/1467-9868.00146.3](https://doi.org/10.1111/1467-9868.00146.3)

- [Rou87] ROUSSEEUW P.: Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* 20, 1 (Nov. 1987), 53–65. URL: [https://doi.org/10.1016/0377-0427\(87\)90125-7](https://doi.org/10.1016/0377-0427(87)90125-7), doi:10.1016/0377-0427(87)90125-7. 3
- [RS00] ROWEIS S., SAUL L.: Nonlinear dimensionality reduction by locally linear embedding. *Science* 290, 5500 (2000), 2323–2326. 3
- [RSJM10a] RAM A., SUNITA J., JALAL A., MANOJ K.: A density based algorithm for discovering density varied clusters in large spatial databases. *International Journal of Computer Applications* 3 (06 2010). doi:10.5120/739-1038. 3
- [RSJM10b] RAM A., SUNITA J., JALAL A., MANOJ K.: A density based algorithm for discovering density varied clusters in large spatial databases. *International Journal of Computer Applications* 3 (06 2010). doi:10.5120/739-1038. 3
- [Saa90] SAAD Y.: Sparskit: A basic toolkit for sparse matrix computations. *Tech.rep.university of Illinois* (1990). 3
- [SH10] SUN S., HUANG R.: An adaptive k-nearest neighbor algorithm. In *2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery* (2010), vol. 1, pp. 91–94. doi:10.1109/FSKD.2010.5569740. 5
- [SSV18] SONDAG M., SPECKMANN B., VERBEEK K.: Stable treemaps via local moves. *IEEE Transactions on Visualization and Computer Graphics* 24, 1 (2018), 729–738. doi:10.1109/TVCG.2017.2745140. 3
- [SVN17] SUN J., VANDIERENDONCK H., NIKOLOPOULOS D. S.: Accelerating graph analytics by utilising the memory locality of graph partitioning. In *2017 46th International Conference on Parallel Processing (ICPP)* (2017), pp. 181–190. doi:10.1109/ICPP.2017.27. 3
- [TSSM16] TAYLOR D., SHAI S., STANLEY N., MUCHA P. J.: Enhanced detectability of community structure in multilayer networks through layer aggregation. *Physical review letters* 116, 22 (June 2016), 228301. doi:10.1103/physrevlett.116.228301. 3
- [TWL09] TANG L., WANG X., LIU H.: Uncovering groups via heterogeneous interaction analysis. In *2009 Ninth IEEE International Conference on Data Mining* (2009), pp. 503–512. doi:10.1109/ICDM.2009.20. 3
- [WCZ16] WANG D., CUI P., ZHU W.: Structural deep network embedding. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2016), KDD ’16, Association for Computing Machinery, p. 1225–1234. URL: <https://doi.org/10.1145/2939672.2939753>, doi:10.1145/2939672.2939753. 3
- [WYL19] WANG H., YANG Y., LIU B.: Gmc: Graph-based multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering PP* (03 2019), 1–1. doi:10.1109/TKDE.2019.2903810. 5
- [YTL*16] YANG H., TANG K., LIU X., XIAO L., XU R., KUMARA S.: A user-centred approach to information visualisation in nano-health. *International Journal of Bioinformatics Research and Applications* 12 (01 2016), 95. doi:10.1504/IJBRA.2016.077122. 3
- [YWL*21] YANG W., WANG X., LU J., DOU W., LIU S.: Interactive steering of hierarchical clustering. *IEEE Transactions on Visualization and Computer Graphics* 27, 10 (2021), 3953–3967. doi:10.1109/TVCG.2020.2995100. 3
- [ZC18] ZHANG M., CHEN Y.: Link prediction based on graph neural networks. *CoRR abs/1802.09691* (2018). URL: <http://arxiv.org/abs/1802.09691>, arXiv:1802.09691. 3
- [ZSS*21] ZHOU Z., SHI C., SHEN X., CAI L., WANG H., LIU Y., ZHAO Y., CHEN W.: Context-aware sampling of large networks via graph representation learning. *IEEE Transactions on Visualization and Computer Graphics* 27, 2 (2021), 1709–1719. doi:10.1109/TVCG.2020.3030440. 3
- [ZWC20] ZHU W., WANG X., CUI P.: *Deep Learning for Learning Graph Representations*. Springer International Publishing, Cham, 2020, pp. 169–210. URL: https://doi.org/10.1007/978-3-030-31756-0_6, doi:10.1007/978-3-030-31756-0_6. 3
- [ZWYY18] ZHANG X., WU L., YAO Z., YU S.: A multi-layer network topology visualization layout based on louvain community detection. In *2018 IEEE Third International Conference on Data Science in Cyberspace (DSC)* (2018), pp. 760–763. doi:10.1109/DSC.2018.00122. 3
- [ZYW*21] ZHAO B.-W., YOU Z.-H., WONG L., ZHANG P., LI H.-Y., WANG L.: Mgrl: Predicting drug-disease associations based on multi-graph representation learning. *Frontiers in Genetics* 12 (04 2021). doi:10.3389/fgene.2021.657182. 3