Applied Soft Computing Journal 164 (2024) 112012

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Diverse joint nonnegative matrix tri-factorization for attributed graph clustering

Arina Mohammadi, Seyed Amjad Seyedi, Fardin Akhlaghian Tab, Rojiar Pir Mohammadiani *

Department of Computer Engineering, University of Kurdistan, Sanandaj, Iran

A R T I C L E I N F O

Keywords:
Attributed graph clustering
Nonnegative matrix factorization
Hilbert–Schmidt independence criterion
Graph regularization

A B S T R A C T

Cluster analysis of attributed graphs is a demanding and challenging task in the analysis of network-structured data. It involves learning node representation by leveraging both node attributes and the topological structure of the graph, aiming to accomplish effective clustering. Typically, existing methods fuse the topological and non-topological information by learning a consensus representation, often resulting in redundancy and overlooking their inherent distinctions. To address this issue, this paper proposes the Diverse Joint Nonnegative Matrix Tri-Factorization (Div-JNMTF), an embedding based model to detect communities in attributed graphs. The novel JNMTF model attempts to extract two distinct node representations from topological and non-topological data. Simultaneously, a diversity regularization technique utilizing the Hilbert–Schmidt Independence Criterion (HSIC) is employed. Its objective is to reduce redundant information in the node representations while encouraging the distinct contributions of both types of information. In addition, two graph regularization terms are introduced to preserve the local structures in the topological and attribute representation spaces. The Div-JNMTF model is optimized by developing an iterative optimization approach. By conducting thorough experiments on four synthetic and eight real-world attributed graph datasets, it has been demonstrated that the proposed model excels in accurately detecting attributed communities and surpasses the performance of existing methods.

1. Introduction

In recent years, the study of networks has gained significant attention, and numerous researchers are actively engaged in developing and implementing algorithms focused on analyzing network structures across various disciplines such as natural sciences, social sciences, and computer sciences [1]. A crucial challenge in network analysis is around the detection and characterization of community structures in networks. Communities can be understood as clusters of nodes which are densely connected within clusters but sparsely connected between groups [2]. This notion of identifying network communities not only aids in gaining understanding of the structural characteristics of the network but also offers practical benefits. For example, communities in social networks tend to share similar interest, which can be leveraged to develop recommendation systems and provide valuable insights [3].

Nevertheless, in most real-life social networks, there is a wealth of information about social actors than just connections between them. In fact, it is quite typical for certain attributes of individuals, such as age, gender, interests, and so on, to be readily accessible. In such cases, the social network is referred to as node-attributed network, recall that the actors are represented via nodes. As stated in Ref. [4], attributes constitute an additional dimension, alongside the structural aspect, when representing social networks. Conventional community detection methods known as structure-aware methods that primarily rely on the relationships between nodes to discover communities. These methods focus exclusively on the network's topology while disregarding any node attributes or additional information in the process. It is evident that methods dealing only with structure or only with attributes fail to utilize the complete information present in a node-attributed social network. Naturally, to overcome this problem, an approach is needed that uses both structure and attribute information when identifying communities. Developing of such methods emerged as a new area in social network analysis [5]. This approach shows great promise, as the combined utilization of structure and attributes is believed to enhance our understanding of social actors and to describe the patterns that form their communities [5,6].

Numerous attributed graph clustering methods have been proposed and they can be broadly categorized into several groups: modifying the objective functions of classical clustering algorithms [7,8],

* Corresponding author.
E-mail addresses: arina.mohammadi@uok.ac.ir (A. Mohammadi), amjadseyedi@uok.ac.ir (S.A. Seyedi), f.akhlaghian@uok.ac.ir (F. Akhlaghian Tab), r.pirmohammadiani@uok.ac.ir (R. Pir Mohammadiani).

https://doi.org/10.1016/j.asoc.2024.112012
Received 14 November 2023; Received in revised form 30 May 2024; Accepted 7 July 2024
Available online 22 July 2024

probabilistic-model-based algorithms [9–12], dynamical system-based approaches [13–15], and Matrix Factorization-based algorithms [16–19].

Nonnegative Matrix Factorization (NMF) has gained significant popularity as a dimensionality reduction method in recent years [20–22]. This technique is well-regarded for its favorable interpretations in psychological and physiological contexts. NMF fundamentally decomposes a high-dimensional input data matrix, $\boldsymbol{X}$, into two nonnegative matrices, namely $\boldsymbol{W}$ (the basis matrix) and $\boldsymbol{H}$ (the coefficient matrix) [20,23]. This decomposition is particularly valuable in identifying the constituent parts of objects. The nonnegative constraints applied to $\boldsymbol{W}$ and $\boldsymbol{H}$ result in a representation based on parts, as they allow only addition operations and prohibit subtraction or combination operations [20,23].

Compared to other algorithms commonly employed for community detection, the methods based on NMF model [20] offer unique advantages, such as greater interpretability and more flexibility in producing community detection outcomes [24–26]. For example, when dealing with a complex network, it is feasible to represent the network as a nonnegative matrix, often referred to as the adjacency matrix. By leveraging the NMF, this attribute matrix can be effectively decomposed, leading to the creation of a membership matrix that assigns nodes to specific communities within the network. Due to the nonnegative constraints imposed on the matrix, each element within the matrix can be understood as the membership strength of the corresponding node within its respective community.

Community detection includes the clustering of objects, particularly nodes, within intricate networks. Approaches such as K-means and spectral clustering have demonstrated their efficacy in tackling node clustering challenges [27]. Consequently, NMF can be seamlessly applied to community detection tasks. Many existing NMF-based graph clustering methods enhance the clustering capability of NMF according to homogeneity assumption [28]. *Wang et al.* [29] introduced the utilization of the NMF algorithm for community detection. Their approach takes into account three types of networks: undirected networks, directed networks, and composite networks. To identify communities within these diverse networks, they presented three distinct techniques: Symmetric NMF (SNMF), Asymmetric NMF (ANMF), and Joint NMF (JNMF). These methods developed to efficiently detect hidden communities in various types of networks [29].

NMF has found extensive application in complex network analysis [30–33]. More recently, some NMF models are proposed for attributed community detection. For example, *Wang et al.* [16] proposed the Semantic Community Identification (SCI), an attributed community detection model based on integrating information of network topologies and information of node attributes under the Symmetric and basic NMF frameworks. SCI extracts community membership matrix from topological structure, and attempts to map the attribute matrix to the extracted membership matrix by a sparse community attribute matrix. *Li et al.* [34] developed the SCI model to the Community Structure Embedding (CDE) model by a community structure embedding matrix instead of simply using the observed network topology. In contrast to SCI, CDE extracts the membership matrix by decomposing attribute matrix.

In order to identify communities within complex social networks using both link and attribute information, Laplacian Joint NMF (LJNMF) [35] joins the symmetric NMF and basic NMF models using a complementary weighting strategy. To enhance the discriminating ability in communities mining, the LJNMF preserves local geometric structure in the cluster space by a graph regularization term. Similar to LJNMF, *Huang et al.* [36] introduced a joint weighted nonnegative factorization method called JWNMF to address the problem of attributed graph clustering. This method combines the Symmetric NMF and weighted NMF terms to integrate both topology and attribute information for clustering attributed graphs. A weighting scheme is incorporated into JWNMF to assign varying importance to attributes within clusters, enabling a more effective clustering.

Maekawa et al. introduced the Non-linear Attribute Graph Clustering (NAGC), which combines Symmetric NMF with Positive Unlabeled Learning [37]. NAGC is a non-linear projection method that connects the latent embedding spaces of the graph's topology and attributes. *Lu et al.* [38] proposed two parameter-free joint NMF methods to detect community structures in attribute networks. Topology and Attribute NMF (TANMF) is a joint basic NMF model that fuses the topological and non-topological information to a shared cluster space. Similarly, Topology and Attribute Symmetrical NMF (TASNMF) extracts cluster matrix from structure and attribute matrices by integrating basic NMF and Symmetric Tri-NMF models. *Berahmand et al.* [39] introduced the AGNMF-AN method, which integrates both structural and attribute information using the Augment Attributed Graph matrix. This approach involves learning the affinity graph to preserve local structure. The AGNMF-AN method offers a comprehensive framework for jointly considering structural information and attribute information for better analysis of attributed networks. *Shang et al.* [40] introduced an attributed graph clustering method, the Latent Representation Learning and graph-regularized Nonnegative Matrix Factorization (LRL-GNMF) model, to intricate relationship between network topology and node attributes. LRL-GNMFT and LRL-GNMFA models are established respectively as models topology-dominated, and models attribute-dominated. Each model decomposes the topological and attribute information by incorporating graph-regularized NMF modules. These modules enable the combination of two clustering results using a transmission matrix, establishing a mapping relation between the results.

Although the mentioned methods provide an interplay between network topology and node attributes, they primarily focus on discovering a shared representation and overlook the distinctions between the two views. Consequently, the learned features tend to be redundant, and only overlapping patterns are identified and preserved. In order to attain a more comprehensive representation, it is essential to develop a method that not only captures shared information but also effectively encodes the unique characteristics of each perspective. By properly addressing the unique characteristics of different views, such a method can provide a holistic representation of the entire dataset, leading to promising outcomes. Recently, *Cui and Li* [41] proposed a multi-view NMF, named non-redundant regularization-based NMF, where, the non-redundant features of multiple views are exploited using a Hilbert Schmidt Independence Criterion (HSIC) regularization [42]. Inspired by the HSIC criterion, this study attempts to explore distinct information contained in structure and attributes information. The proposed method extracts more distinct representation by considering differences in the structure and attributes information. The key idea is that minimizing overlaps between the structural and attribute representations can further accentuate the differences and extract more distinct representations capturing diverse perspectives. In this paper, an NMF-based model named Diverse Joint Nonnegative Matrix Tri-Factorization (Div-JNMTF) is proposed. The proposed JNMTF model joins structure factorization and attribute factorization by a shared latent correlation matrix, and extracts diverse node representations. The regularized JNMTF technique utilizes the Hilbert–Schmidt independence criterion (HSIC) to reduce the redundant representation of both topological and non-topological data, while promoting the preservation of distinct information. Moreover, this method incorporates two graph regularization terms into the objective function to maintain the local structures of the input graph and its attributes within the latent representation. Finally, the acquired representations from both views are concatenated to create a final representation of the data. This representation by combining the complementary information gathered from both views, provides a comprehensive understanding of the data. The primary contributions of this method include:

- The proposed Joint Nonnegative Matrix Tri-Factorization model by factorizing the topological and non-topological inputs and extracting a common latent correlation matrix, maps the structure and attribute views to different low-dimensional representations.
- To avoid extracting redundant features, a HSIC penalty term is imposed which increases the diversity between topological a and non-topological representations.
- To preserve the local structure of both information in the latent space, dual graph regularizations are incorporated in this model.
- The proposed Div-JNMTF model with dual local structure preservation, learns diverse and non-redundant representations from an attributed network. Experimental results on eight real-world attributed networks demonstrates the effectiveness of the proposed model.

The rest of this paper is organized as follows. Section 2 briefly reviews the basic concepts of NMF models and HSIC regularization. Section 3 introduces the proposed Div-JNMTF model and experimental results are presented in Section 4. Finally, Section 5 concludes the paper and presents the directions for future works.

2. Background

This section reviews relevant models and key concepts that provide context for the proposed method. Specifically, it introduces several NMF models that are leveraged in the proposed model, including standard NMF, Nonnegative Matrix Tri-Factorization, Symmetric NMF, and Symmetric Nonnegative Matrix Tri-Factorization. The section also briefly explains the Hilbert–Schmidt Independence Criterion, which is utilized as a regularization in the proposed model.

2.1. Nonnegative matrix factorization

Lee and Seung [20] introduced the Nonnegative Matrix Factorization (NMF) model in 1999. It aims to estimate the optimal local parameters $\boldsymbol{W}$ and $\boldsymbol{H}$ by minimizing a non-convex loss between the original data matrix $\boldsymbol{X}$ and approximate representation $\boldsymbol{W}\boldsymbol{H}$. NMF has become widely popular in the field of machine learning and is widely employed as a representation learning method due to its favorable interpretability [43–48]. The standard formula for NMF is provided below:

$$\min_{\boldsymbol{W},\boldsymbol{H}} \|\boldsymbol{X} - \boldsymbol{W}\boldsymbol{H}^\top\|_F^2, \quad \text{s.t. } \boldsymbol{W},\boldsymbol{H} \geq 0, \tag{1}$$

where $\boldsymbol{X} \in \mathbb{R}_+^{d\times n}$ is a given matrix to be decomposed, $\boldsymbol{W} \in \mathbb{R}_+^{d\times k}$ is basis matrix, $\boldsymbol{H} \in \mathbb{R}_+^{n\times k}$ is coefficient matrix, and $k \ll \min(d, n)$.

2.2. Nonnegative matrix tri-factorization

Nonnegative Matrix Tri-Factorization (NMTF) [49] is an extension of NMF for co-clustering that supposes different numbers of clusters for samples and attributes. This approach introduces a transfer matrix $\boldsymbol{B} \in \mathbb{R}_+^{k_1\times k_2}$ that captures the relationship between the clusters of vertices and attributes. NMTF decomposes the attribute matrix as follow:

$$\min_{\boldsymbol{W},\boldsymbol{B},\boldsymbol{H}} \|\boldsymbol{X} - \boldsymbol{W}\boldsymbol{B}\boldsymbol{H}^\top\|_F^2, \quad \text{s.t. } \boldsymbol{W},\boldsymbol{B},\boldsymbol{H} \geq 0, \tag{2}$$

where $\boldsymbol{W} \in \mathbb{R}_+^{d\times k_1}$ is an attribute factor matrix with k_1 factors and $\boldsymbol{H} \in \mathbb{R}^{n\times k_2}$ is a membership matrix. This method extracts linear relationships between sample clusters and attribute clusters.

2.3. Symmetric nonnegative matrix factorization

NMF is a powerful method for generating low-rank approximation of a nonnegative matrix and has demonstrated potential as a representation method; however, it is not a direct clustering method. To overcome this limitation, Symmetric NMF (SNMF) [28,29] introduces a comprehensive understanding of the strengths and limitations of utilizing NMF in the context of graph clustering. SNMF enhances part-based representation by preserving nonnegativity in the assignment matrix and incorporating a contrastive mechanism, evolving it into a clustering model. Unlike NMF, SNMF operates based on a similarity measure between data points and factorizes a symmetric matrix that contains pairwise similarity values. The objective function for SNMF is formulated as follows:

$$\min_{\boldsymbol{H}} \|\boldsymbol{A} - \boldsymbol{H}\boldsymbol{H}^\top\|_F^2, \quad \text{s.t. } \boldsymbol{H} \geq 0. \tag{3}$$

2.4. Symmetric nonnegative matrix tri-factorization

Symmetric Nonnegative Matrix Tri-Factorization (SNMTF) [29] extends NMF by the decomposing a matrix into three matrices and introduces an additional symmetric latent matrix $\boldsymbol{B}$, which captures the interaction patterns among clusters. Consequently, SNMTF has been employed in community detection [29,50] to identify communities and understand their interactions. The objective function for SNMTF is formulated as follows:

$$\min_{\boldsymbol{B},\boldsymbol{H}} \|\boldsymbol{A} - \boldsymbol{H}\boldsymbol{B}\boldsymbol{H}^\top\|_F^2, \quad \text{s.t. } \boldsymbol{H},\boldsymbol{B} \geq 0. \tag{4}$$

2.5. Hilbert–Schmidt independence criterion

Consider two random variables $(\boldsymbol{x}, \boldsymbol{y})$ drawn from two different sample spaces $\mathcal{X}$ and $\mathcal{Y}$ that share a joint distribution. We define a mapping from $\boldsymbol{x} \in \mathcal{X}$ to a kernel space $\mathcal{F}$, denoted as $\phi(\boldsymbol{x})$, such that the inner product between vectors in this kernel space can be expressed using a kernel function $k_1(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle\phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j)\rangle$, where $\langle\cdot\,,\,\cdot\rangle$ represents the inner product. Similarly, we define another mapping from $\boldsymbol{y} \in \mathcal{Y}$ to a second kernel space $\mathcal{G}$, denoted as $\varphi(\boldsymbol{y})$, where the inner product between vectors in this kernel space is given by $k_2(\boldsymbol{y}_i, \boldsymbol{y}_j)$. In this context, the cross-covariance can be described as follows:

$$C_{\boldsymbol{xy}} = E_{\boldsymbol{xy}}[(\phi(\boldsymbol{x}) - \mu_{\boldsymbol{x}})\otimes(\varphi(\boldsymbol{y}) - \mu_{\boldsymbol{y}})], \tag{5}$$

where $\otimes$ is the tensor product, $\mu_{\boldsymbol{x}} = E(\phi(\boldsymbol{x})), \mu_{\boldsymbol{y}} = E[\varphi(\boldsymbol{y})]$, and $E(.)$ is the expectation operator. As indicated to [42], the HSIC is defined in the following manner:

Definition 1. Given $\mathcal{F}$ and $\mathcal{G}$ are two separable Reproducing Kernel Hilbert Spaces (RKHSs), and $\boldsymbol{P}_{\boldsymbol{xy}}$ is a joint distribution. The HSIC, defined with respect to the cross-covariance operator $C_{\boldsymbol{xy}}$ and employing a squared Hilbert Schmidt norm, takes the following form:

$$\mathrm{HSIC}(P_{\boldsymbol{xy}}, \mathcal{F}, \varphi) := \|C_{\boldsymbol{xy}}\|_{\mathrm{HS}}^2. \tag{6}$$

The Hilbert–Schmidt norm of a matrix is denoted as $\|.\|_{\mathrm{HS}}$. To compute the Hilbert–Schmidt norm of a given matrix $\boldsymbol{B}$, the following expression is used:

$$\|\boldsymbol{B}\|_{\mathrm{HS}} = \sqrt{\sum_{i,j} B_{i,j}^2}. \tag{7}$$

Since the joint distribution $P_{\boldsymbol{xy}}$ is often unknown or challenging to estimate, researchers commonly resort to using the empirical version of HSIC. Given n observations drawn from the joint distribution $p_{\boldsymbol{xy}}, \mathcal{Z} := \{(x_1, y_1), \dots, (x_n, y_n)\}$; the empirical HSIC can be expressed as follows:

$$\mathrm{HSIC}(\mathcal{Z}, \mathcal{F}, \varphi) := \frac{1}{(\boldsymbol{n}-1)^2}\mathrm{Tr}(\boldsymbol{K}_1\boldsymbol{Q}\boldsymbol{K}_2\boldsymbol{Q}), \tag{8}$$

where $\mathrm{Tr}(\cdot)$ is the matrix trace, $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$ are Gram matrices with $K_{1,ij} = k_1(x_i, x_j)$ and $K_{2,ij} = k_2(y_i, y_j)$. $\boldsymbol{Q} = \boldsymbol{I} - \frac{1}{n}\boldsymbol{e}\boldsymbol{e}^\top$ is a centering matrix, $\boldsymbol{I}$ represents an identity matrix, and $\boldsymbol{e}$ is an all-one column vector. Further details about HSIC can be found in the research [42].

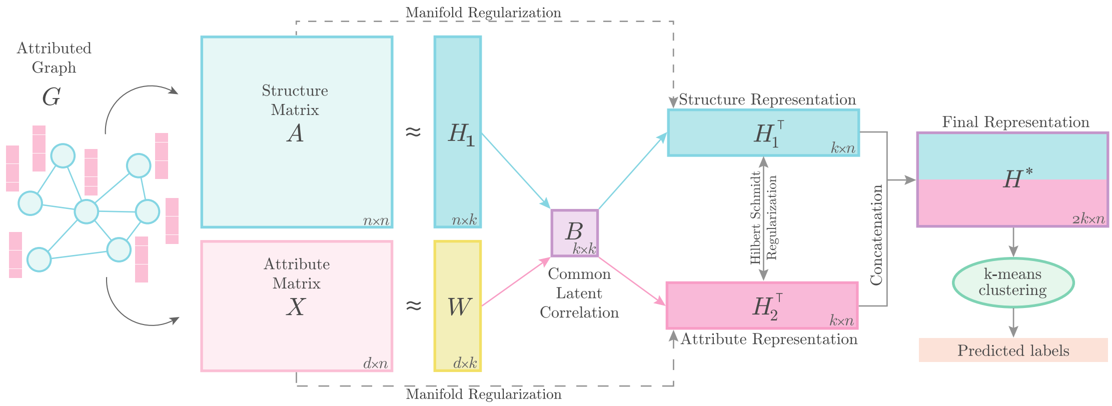


Fig. 1. Schematic Representation of the Proposed Div-JNMTF Model, where n, d, and k are the number of nodes, attribute dimension, and latent factor, respectively.

3. Proposed method

In this section, we propose the Diverse Joint Nonnegative Matrix Tri-Factorization (Div-JNMTF) model for clustering in attributed networks. In contrast to other Joint NMF models which map the structure and attribute matrices to a shared representation space, the basic proposed model tri-factorizes structure and attribute matrices to a shared latent correlation matrix and two different representations. To extract more distinct and diverse representations, the proposed Div-JNMTF model minimizes the HSIC between structure and attribute representations. In addition, dual graph regularization terms are added to the model in order to preserve intrinsic manifold of structure and attribute data by exploiting the prior similarity information. Finally, we present the multiplicative updating rules of the Div-JNMTF model. Fig. 1 provides a schematic representation of the Div-JNMTF model.

3.1. Basic model

To community mining in complex social networks that incorporate both link information and attribute information, existing joint NMF models [29] map both information into a shared representation space. This mapping extracts common information, and results in ignoring the exclusive information of each view. In contrast to these joint models, this paper proposes a modified joint factorization model to prevent exploiting redundant features, and attempts to learn non-redundant complementary representation of views by factorizing their common information. The proposed JNMTF integrates the SNMTF (4) and NMTF (2) models for factorizing the structure and attribute matrices, respectively. The objective function of the proposed JNMTF is defined as

$$\min_{\boldsymbol{H}_1,\boldsymbol{H}_2,\boldsymbol{B},\boldsymbol{W}} \|\boldsymbol{A}-\boldsymbol{H}_1\boldsymbol{B}\boldsymbol{H}_1^\top\|_F^2+\alpha\|\boldsymbol{X}-\boldsymbol{W}\boldsymbol{B}\boldsymbol{H}_2^\top\|_F^2, \text{ s.t. } \boldsymbol{H}_1,\boldsymbol{H}_2,\boldsymbol{B},\boldsymbol{W}\geq 0, \tag{9}$$

where $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ are structure and attribute representations, respectively, $\boldsymbol{B}$ is a shared latent correlation matrix, and α is a trade-off hyper-parameter that determines the contribution of each view. In this model, factorizing common information $\boldsymbol{B}$ from link and attribute data leads to extract more distinct information in $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ representations.

3.2. HSIC regularization

To explore the complementary and non-redundant information in attributed graphs, it is necessary to minimize the redundancy in the semantic representations derived from both structural and attribute information. In Eq. (9), the matrices $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ represent the low-dimensional representations of different views obtained through Joint NMTF. However, these representations may still contain some redundant statistical information shared by both information sources, without adequately considering the non-overlapped features present in the topological and non-topological views. Suitable leveraging both shared and distinct information between two views can lead to a comprehensive description of the attributed graph. In order to capture a comprehensive description of the attributed graph by leveraging the differences between the views, it is important to reduce redundancy in the representations and promote the incorporation of complementary information. To achieve this, we add a diversity regularization based on Hilbert–Schmidt Independence Criterion (HSIC) [41] to the basic JNMTF model (9). The resulting formulation, referred to as Div-JNMTF, is as follows:

$$\begin{aligned}\min_{\boldsymbol{H}_1,\boldsymbol{H}_2,\boldsymbol{B},\boldsymbol{W}} &\|\boldsymbol{A}-\boldsymbol{H}_1\boldsymbol{B}\boldsymbol{H}_1^\top\|_F^2+\alpha\|\boldsymbol{X}-\boldsymbol{W}\boldsymbol{B}\boldsymbol{H}_2^\top\|_F^2+\lambda_1\mathrm{HSIC}(\boldsymbol{H}_1,\boldsymbol{H}_2),\\ &\text{s.t. } \boldsymbol{H}_1,\boldsymbol{H}_2,\boldsymbol{B},\boldsymbol{W}\geq 0,.\end{aligned} \tag{10}$$

The third term in the equation, represents the diversity regularization, which is defined using the HSIC [41] on the representations obtained from two views. Minimizing this term can promote the semantic variety of the representations derived from the both views. For simplicity, the inner product kernels are adopted for the HSIC (8), that are, $\boldsymbol{K}_1=\boldsymbol{H}_1\boldsymbol{H}_1^\top$ and $\boldsymbol{K}_2=\boldsymbol{H}_2\boldsymbol{H}_2^\top$. Then, Eq. (10) can be rewritten as

$$\begin{aligned}\min_{\boldsymbol{H}_1,\boldsymbol{H}_2,\boldsymbol{B},\boldsymbol{W}} &\|\boldsymbol{A}-\boldsymbol{H}_1\boldsymbol{B}\boldsymbol{H}_1^\top\|_F^2+\alpha\|\boldsymbol{X}-\boldsymbol{W}\boldsymbol{B}\boldsymbol{H}_2^\top\|_F^2\\ &+\lambda_1\mathrm{Tr}(\boldsymbol{H}_1\boldsymbol{H}_1^\top\boldsymbol{Q}\boldsymbol{H}_2\boldsymbol{H}_2^\top\boldsymbol{Q}), \text{ s.t. } \boldsymbol{H}_1,\boldsymbol{H}_2,\boldsymbol{B},\boldsymbol{W}\geq 0,\end{aligned} \tag{11}$$

where the parameter λ_1 in the above objective function, absorbs the scaling factor $1/(n-1)^2$.

3.3. Dual graph regularizations

NMF leverages nonnegative constraints to acquire a parts-based representation. However, its learning process occurs solely within the Euclidean space, thereby overlooking the inherent geometric and discriminatory structure of the data space, which is crucial for practical applications [51]. To overcome this drawback, we propose a solution by introducing dual graph regularization terms that consider the local structure of both the graph and attribute data. Specifically, we define the manifold learning [51] term for the structural data as follows:

$$\begin{aligned}\mathcal{R}_S &= \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\|\boldsymbol{h}_i^{(1)}-\boldsymbol{h}_j^{(1)}\|^2 S_{i,j}^{(1)}\\ &=\mathrm{Tr}(\boldsymbol{H}_1^\top\boldsymbol{D}_1\boldsymbol{H}_1)-\mathrm{Tr}(\boldsymbol{H}_1^\top\boldsymbol{S}_1\boldsymbol{H}_1)=\mathrm{Tr}(\boldsymbol{H}_1^\top\boldsymbol{L}_1\boldsymbol{H}_1),\end{aligned} \tag{12}$$

where $S_{ij}^{(1)}$ denotes the (i,j)-th element of the similarity matrix $\boldsymbol{S}_1$ obtained from graph $\boldsymbol{A}$, the Laplacian matrix $\boldsymbol{L}_1=\boldsymbol{D}_1-\boldsymbol{S}_1$, and diagonal matrix $\boldsymbol{D}_1$ calculated by $D_{ii}^{(1)}=\sum_{j=1}^{n}S_{ij}^{(1)}$. The local geometric structure can be accurately represented by constructing a nearest neighbor graph of samples. We create a graph with n vertices, where each vertex corresponds to a specific sample. To build the graph, for each sample $\boldsymbol{x}_j$, we identify its p nearest neighbors and establish edges between $\boldsymbol{x}_j$ and its neighboring samples. The similarity matrix $\boldsymbol{S}_1$ is constructed by the cosine similarity measure as follow:

$$S^{(1)}_{i,j} = \frac{\boldsymbol{a}_i^\top \boldsymbol{a}_j}{\|\boldsymbol{a}_i\|.\|\boldsymbol{a}_j\|}, \tag{13}$$

where $\boldsymbol{a}_i$ and $\boldsymbol{a}_j$ are the ith and jth columns of the adjacency matrix $\boldsymbol{A}$. Similarly, the manifold learning term for attribute data is defined as follow:

$$\begin{aligned}\mathcal{R}_A &= \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\|\boldsymbol{h}_i^{(2)} - \boldsymbol{h}_j^{(2)}\|^2 S^{(2)}_{i,j} \\ &= \mathrm{Tr}(\boldsymbol{H}_2^\top \boldsymbol{D}_2 \boldsymbol{H}_2) - \mathrm{Tr}(\boldsymbol{H}_2{}^\top \boldsymbol{S}_2 \boldsymbol{H}_2) = \mathrm{Tr}(\boldsymbol{H}_2^\top \boldsymbol{L}_2 \boldsymbol{H}_2),\end{aligned} \tag{14}$$

where $S^{(2)}_{ij}$ denotes the (i,j)-th element of the similarity matrix $\boldsymbol{S}_2$ obtained from attribute matrix $\boldsymbol{X}$, the Laplacian matrix $\boldsymbol{L}_2 = \boldsymbol{D}_2 - \boldsymbol{S}_2$, and diagonal matrix $\boldsymbol{D}_2$ calculated by $D^{(2)}_{ii} = \sum_{j=1}^{n} S^{(2)}_{ij}$. The similarity matrix $\boldsymbol{S}_2$ is constructed by the cosine function to measure the similarity of node attribute $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ in the attribute space as follow:

$$S^{(2)}_{i,j} = \frac{\boldsymbol{x}_i^\top \boldsymbol{x}_j}{\|\boldsymbol{x}_i\|.\|\boldsymbol{x}_j\|}. \tag{15}$$

By minimizing Eqs. (12) and (14), if node i and node j have large similarity values in structure or attribute spaces, their representation pairs $\boldsymbol{h}_i^{(v)}$ and $\boldsymbol{h}_j^{(v)}$ are set to be as close as possible. Finally, by adding these regularizations to (11), the final objective function of Div-JNMTF model is obtained as

$$\begin{aligned}\mathcal{L} = \min_{\boldsymbol{H}_1,\boldsymbol{H}_2,\boldsymbol{B},\boldsymbol{W}} &\|\boldsymbol{A} - \boldsymbol{H}_1\boldsymbol{B}\boldsymbol{H}_1^\top\|_F^2 + \alpha\|\boldsymbol{X} - \boldsymbol{W}\boldsymbol{B}\boldsymbol{H}_2^\top\|_F^2 \\ &+ \lambda_1\mathrm{Tr}(\boldsymbol{H}_1\boldsymbol{H}_1^\top\boldsymbol{Q}\boldsymbol{H}_2\boldsymbol{H}_2^\top\boldsymbol{Q}) + \lambda_2[\mathrm{Tr}(\boldsymbol{H}_1^\top\boldsymbol{L}_1\boldsymbol{H}_1) + \mathrm{Tr}(\boldsymbol{H}_2^\top\boldsymbol{L}_2\boldsymbol{H}_2)], \\ &\text{s.t. } \boldsymbol{H}_1, \boldsymbol{H}_2, \boldsymbol{B}, \boldsymbol{W} \geq 0.\end{aligned} \tag{16}$$

Since the objective function (16) is solved, the resulting representations $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ are obtained, and their concatenation is used as the final combined representation $\boldsymbol{H}^* = [\boldsymbol{H}_1; \boldsymbol{H}_2]^\top \in \mathbb{R}^{2k\times n}$.

3.4. Optimization

This section provides a comprehensive explanation of the optimization process for the objective function of Div-JNMTF. It is important to note that objective function (16) is non-convex with respect to the variables $\boldsymbol{H}_1$, $\boldsymbol{H}_2$, $\boldsymbol{W}$, and $\boldsymbol{B}$, making it challenging to find the global minimum. By incorporating non-redundancy and manifold regularization, we have taken steps towards mitigating the challenges of non-convex optimization and improving the generalization capabilities of our model. Empirical evidence from analogous studies underscores the efficacy of our optimization strategy. For instance, [52–54] have demonstrated the effectiveness of similar approaches in optimizing non-convex objective functions, achieving convergence to promising solutions in various applications. The empirical results in Section 4.7 demonstrate the robust convergence of the optimization process of the proposed method, illustrating its effectiveness in navigating the non-convex landscape of the objective function and reaching acceptable solutions. Moreover, it is worth highlighting that function (16) becomes convex for each factor when the other factors are held constant. To address this problem, an alternative approach is adopted where each factor is optimized while the remaining factors are fixed. The specific details of this optimization scheme are described in the following:

- *Updating rule for the structure representation matrix* $\boldsymbol{H}_1$.
  By fixing all the variables except for $\boldsymbol{H}_1$, the objective function in (16) is reduced to:

$$\begin{aligned}\min_{\boldsymbol{H}_1} \mathcal{L}(\boldsymbol{H}_1) = \|\boldsymbol{A} - \boldsymbol{H}_1\boldsymbol{B}\boldsymbol{H}_1^\top\|_F^2 + \lambda_1\mathrm{Tr}(\boldsymbol{H}_1\boldsymbol{H}_1^\top\boldsymbol{Q}\boldsymbol{H}_2\boldsymbol{H}_2^\top\boldsymbol{Q}) \\ + \lambda_2\mathrm{Tr}(\boldsymbol{H}_1^\top\boldsymbol{L}_1\boldsymbol{H}_1), \text{ s.t. } \boldsymbol{H}_1 \geq 0.\end{aligned} \tag{17}$$

To solve (17), we introduce a Lagrangian multiplier matrix $\boldsymbol{\Theta}$ to enforce the nonnegative constraints on $\boldsymbol{H}_1$, resulting in the following equivalent objective function:

$$\begin{aligned}\min_{\boldsymbol{H}_1,\boldsymbol{\Theta}} \mathcal{L}(\boldsymbol{H}_1,\boldsymbol{\Theta}) =& \|\boldsymbol{A} - \boldsymbol{H}_1\boldsymbol{B}\boldsymbol{H}_1^\top\|_F^2 + \lambda_1\mathrm{Tr}(\boldsymbol{H}_1\boldsymbol{H}_1^\top\boldsymbol{Q}\boldsymbol{H}_2\boldsymbol{H}_2^\top\boldsymbol{Q}) \\ &+ \lambda_2\mathrm{Tr}(\boldsymbol{H}_1^\top\boldsymbol{L}_1\boldsymbol{H}_1) - \mathrm{Tr}(\boldsymbol{\Theta}\boldsymbol{H}_1^\top),\end{aligned} \tag{18}$$

which can be further rewritten as follows:

$$\begin{aligned}\min_{\boldsymbol{H}_1,\boldsymbol{\Theta}} \mathcal{L}(\boldsymbol{H}_1,\boldsymbol{\Theta}) &= \mathrm{Tr}(-2\boldsymbol{A}^\top\boldsymbol{H}_1\boldsymbol{B}\boldsymbol{H}_1^\top + \boldsymbol{H}_1\boldsymbol{B}^\top\boldsymbol{H}_1^\top\boldsymbol{H}_1\boldsymbol{B}\boldsymbol{H}_1^\top) \\ &+ \lambda_1\mathrm{Tr}(\boldsymbol{H}_1\boldsymbol{H}_1^\top\boldsymbol{Q}\boldsymbol{H}_2\boldsymbol{H}_2^\top\boldsymbol{Q}) + \lambda_2\mathrm{Tr}(\boldsymbol{H}_1^\top\boldsymbol{L}_1\boldsymbol{H}_1) - \mathrm{Tr}(\boldsymbol{\Theta}\boldsymbol{H}_1^\top).\end{aligned} \tag{19}$$

By setting the partial derivative of $\mathcal{L}(\boldsymbol{H}_1,\boldsymbol{\Theta})$ with respect to $\boldsymbol{H}_1$ to $\boldsymbol{0}$, we have:

$$\begin{aligned}\boldsymbol{\Theta} =& -4\boldsymbol{A}^\top\boldsymbol{H}_1\boldsymbol{B} + 4\boldsymbol{H}_1\boldsymbol{B}^\top\boldsymbol{H}_1^\top\boldsymbol{H}_1\boldsymbol{B} + 2\lambda_1\boldsymbol{Q}\boldsymbol{H}_2\boldsymbol{H}_2^\top\boldsymbol{Q}\boldsymbol{H}_1 \\ &+ 2\lambda_2\boldsymbol{D}_2\boldsymbol{H}_1 - 2\lambda_2\boldsymbol{S}_1\boldsymbol{H}_1.\end{aligned} \tag{20}$$

From the complementary slackness condition of the Karush–Kuhn–Tucker (KKT) conditions [23], we obtain:

$$\boldsymbol{H}_1 \odot \boldsymbol{\Theta} = \boldsymbol{0}, \tag{21}$$

where $\odot$ denotes the element-wise product. Eq. (21) is the fixed point equation that the solution must satisfy at convergence. By solving this equation, we derive the following updating rule for $\boldsymbol{H}_1$:

$$\boldsymbol{H}_1 \leftarrow \boldsymbol{H}_1 \odot \left(\frac{\boldsymbol{A}^\top\boldsymbol{H}_1\boldsymbol{B} + \frac{\lambda_1}{2}(\boldsymbol{Q}_p\boldsymbol{K}_2\boldsymbol{Q}_n\boldsymbol{H}_1 + \boldsymbol{Q}_n\boldsymbol{K}_2\boldsymbol{Q}_p\boldsymbol{H}_1) + \frac{\lambda_2}{2}\boldsymbol{S}_1\boldsymbol{H}_1}{\boldsymbol{H}_1\boldsymbol{B}^\top\boldsymbol{H}_1^\top\boldsymbol{H}_1\boldsymbol{B} + \frac{\lambda_1}{2}(\boldsymbol{Q}_n\boldsymbol{K}_2\boldsymbol{Q}_n\boldsymbol{H}_1 + \boldsymbol{Q}_p\boldsymbol{K}_2\boldsymbol{Q}_p\boldsymbol{H}_1) + \frac{\lambda_2}{2}\boldsymbol{D}_1\boldsymbol{H}_1}\right)^{\frac{1}{4}} \tag{22}$$

- *Updating rule for the attribute representation matrix* $\boldsymbol{H}_2$.
  By fixing all the variables except for $\boldsymbol{H}_2$, the objective function in (16) is reduced to:

$$\begin{aligned}\min_{\boldsymbol{H}_2} \mathcal{L}(\boldsymbol{H}_2) = \|\boldsymbol{X} - \boldsymbol{W}\boldsymbol{B}\boldsymbol{H}_2^\top\|_F^2 + \lambda_1\mathrm{Tr}(\boldsymbol{H}_1\boldsymbol{H}_1^\top\boldsymbol{Q}\boldsymbol{H}_2\boldsymbol{H}_2^\top\boldsymbol{Q}) \\ + \lambda_2\mathrm{Tr}(\boldsymbol{H}_2^\top\boldsymbol{L}_2\boldsymbol{H}_2), \text{ s.t. } \boldsymbol{H}_2 \geq 0.\end{aligned} \tag{23}$$

The gradient of the objective in (23) w.r.t. $\boldsymbol{H}_2$ is

$$\begin{aligned}\nabla_{\boldsymbol{H}_2} =& -2\alpha\boldsymbol{X}^\top\boldsymbol{W}\boldsymbol{B} + 2\alpha\boldsymbol{H}_2\boldsymbol{B}^\top\boldsymbol{W}^\top\boldsymbol{W}\boldsymbol{B} \quad (24)\\ &- 2\lambda_1(\boldsymbol{Q}_p\boldsymbol{K}_1\boldsymbol{Q}_n\boldsymbol{H}_2 + \boldsymbol{Q}_n\boldsymbol{K}_1\boldsymbol{Q}_p\boldsymbol{H}_2) \\ &+ 2\lambda_1(\boldsymbol{Q}_n\boldsymbol{K}_1\boldsymbol{Q}_n\boldsymbol{H}_2 + \boldsymbol{Q}_p\boldsymbol{K}_1\boldsymbol{Q}_p\boldsymbol{H}_2) - 2\lambda_2\boldsymbol{S}_2\boldsymbol{H}_2 + 2\lambda_2\boldsymbol{D}_2\boldsymbol{H}_2. \quad (25)\end{aligned}$$

Following similar process of the updating rule for $\boldsymbol{H}_1$, the updating rule for $\boldsymbol{H}_2$ is formulated as follows:

$$\boldsymbol{H}_2 \leftarrow \boldsymbol{H}_2 \odot \frac{\alpha\boldsymbol{X}^\top\boldsymbol{W}\boldsymbol{B} + \lambda_1(\boldsymbol{Q}_p\boldsymbol{K}_1\boldsymbol{Q}_n\boldsymbol{H}_2 + \boldsymbol{Q}_n\boldsymbol{K}_1\boldsymbol{Q}_p\boldsymbol{H}_2) + \lambda_2\boldsymbol{S}_2\boldsymbol{H}_2}{\alpha\boldsymbol{H}_2\boldsymbol{B}^\top\boldsymbol{W}^\top\boldsymbol{W}\boldsymbol{B} + \lambda_1(\boldsymbol{Q}_n\boldsymbol{K}_1\boldsymbol{Q}_n\boldsymbol{H}_2 + \boldsymbol{Q}_p\boldsymbol{K}_1\boldsymbol{Q}_p\boldsymbol{H}_2) + \lambda_2\boldsymbol{D}_2\boldsymbol{H}_2}. \tag{26}$$

- *Updating rule for the feature matrix* $\boldsymbol{W}$.
  To optimize $\boldsymbol{W}$, we in fact seek to optimize the following objective function:

$$\min_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{W}) = \|\boldsymbol{X} - \boldsymbol{W}\boldsymbol{B}\boldsymbol{H}_2^\top\|_F^2, \text{ s.t. } \boldsymbol{W} \geq 0. \tag{27}$$

The gradient of the objective in (27) w.r.t. $\boldsymbol{W}$ is

$$\nabla_{\boldsymbol{W}} = -2\boldsymbol{X}\boldsymbol{H}_2\boldsymbol{B}^\top + 2\boldsymbol{W}\boldsymbol{B}\boldsymbol{H}_2^\top\boldsymbol{H}_2\boldsymbol{B}^\top. \tag{28}$$

Similar to other factors, $\boldsymbol{W}$ can be updated by

$$\boldsymbol{W} \leftarrow \boldsymbol{W} \odot \frac{\boldsymbol{X}\boldsymbol{H}_2\boldsymbol{B}^\top}{\boldsymbol{W}\boldsymbol{B}\boldsymbol{H}_2^\top\boldsymbol{H}_2\boldsymbol{B}^\top}. \tag{29}$$

- *Updating rule for the correlation matrix* $\boldsymbol{B}$.
  To optimize $\boldsymbol{B}$, we in fact seek to optimize the following objective function:

$$\min_{\boldsymbol{B}} \mathcal{L}(\boldsymbol{B}) = \|\boldsymbol{A} - \boldsymbol{H}_1\boldsymbol{B}\boldsymbol{H}_1^\top\|_F^2 + \|\boldsymbol{X} - \boldsymbol{W}\boldsymbol{B}\boldsymbol{H}_2^\top\|_F^2, \text{ s.t. } \boldsymbol{B} \geq 0. \tag{30}$$

The gradient of the objective in (30) w.r.t. $\boldsymbol{B}$ is

$$\nabla_{\boldsymbol{B}} = -2\boldsymbol{H}_1^\top\boldsymbol{A}\boldsymbol{H}_1 + 2\boldsymbol{H}_1^\top\boldsymbol{H}_1\boldsymbol{B}\boldsymbol{H}_1^\top\boldsymbol{H}_1 - 2\alpha\boldsymbol{W}^\top\boldsymbol{X}\boldsymbol{H}_2 + 2\alpha\boldsymbol{W}^\top\boldsymbol{W}\boldsymbol{B}\boldsymbol{H}_2^\top\boldsymbol{H}_2. \tag{31}$$

Finally, $\boldsymbol{B}$ can be updated by

$$\boldsymbol{B} \leftarrow \boldsymbol{B} \odot \frac{\boldsymbol{H}_1^\top\boldsymbol{A}\boldsymbol{H}_1 + \alpha\boldsymbol{W}^\top\boldsymbol{X}\boldsymbol{H}_2}{\boldsymbol{H}_1^\top\boldsymbol{H}_1\boldsymbol{B}\boldsymbol{H}_1^\top\boldsymbol{H}_1 + \alpha\boldsymbol{W}^\top\boldsymbol{W}\boldsymbol{B}\boldsymbol{H}_2^\top\boldsymbol{H}_2}. \tag{32}$$

Until now, we have all the updating rules done. The overall optimization process of Div-JNMTF is outlined in Algorithm 1.

Algorithm 1 Diverse Joint Nonnegative Matrix Tri-Factorization (Div-JNMTF)

Input: Structure matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, Attribute matrix $\boldsymbol{X} \in \mathbb{R}^{d\times n}$, Regularization parameters α, λ_1, and λ_2, latent factor k;
Output: Node representation $\boldsymbol{H}^*$.

1: **Initialize** $\boldsymbol{H}_1, \boldsymbol{H}_2, \boldsymbol{W}, \boldsymbol{B}$ randomly; $t = 0$;
2: Construct similarity graphs $\boldsymbol{S}^{(1)}$ and $\boldsymbol{S}^{(2)}$ according to (13) and (15), respectively;
3: **while** $t <$ MaxIteration or $|\mathcal{L}^{(t-1)} - \mathcal{L}^{(t)}| < \epsilon$ according to (16) **do**
4: Update $\boldsymbol{H}_1$ by (22);
5: Update $\boldsymbol{H}_2$ by (26);
6: Update $\boldsymbol{W}$ by (29);
7: Update $\boldsymbol{B}$ by (32);
8: $t = t + 1$;
9: **end while**
10: **Return** $\boldsymbol{H}^* = [\boldsymbol{H}_1; \boldsymbol{H}_2]^\top$.

3.5. Analysis of computational complexity

In this section, we analyze the time complexity of the proposed Div-JNMTF algorithm. The construction of the kNN similarity graphs $\boldsymbol{S}^{(1)}$ and $\boldsymbol{S}^{(2)}$ using the kd-tree data structure contributes a one-time complexity of $O(nlogn)$ for preprocessing the data, where n is the number of nodes. The time complexity is dominated by the matrix operations involved in updating the factor matrices $\boldsymbol{H}_1$, $\boldsymbol{H}_2$, $\boldsymbol{W}$, and $\boldsymbol{B}$ over multiple iterations. It is worth mentioning that producing the centering matrix $\boldsymbol{Q}$, which is a rank-one update using the Sherman–Morrison formula [55], has a complexity of $O(n^2)$. Therefore, the updates for $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ (Eqs. (22) and (26)) involve multiplications of matrices, which have a time complexity of $O(n^2k)$ and $O(n^2k + dnk)$ per iteration, where k is the latent factor dimensionality, and d is the number of attributes. The updates for $\boldsymbol{W}$ and $\boldsymbol{B}$ (Eqs. (29) and (32)) also involve dense matrix multiplications with a complexity of $O(dnk)$ and $O(n^2k)$ per iteration, respectively. To calculate the overall complexity, we sum up the complexities of each operation over all iterations. If the number of iterations required for convergence as t, then, the overall time complexity of the Div-JNMTF model is $O(nlogn+tn^2k+tdnk)$. In Table 1, we list the time complexity of related methods, and it is evident that the time complexity of the Div-JNMTF model is comparable with state-of-the-art methods. The complexity analysis reveals that the proposed method achieves similar time complexity to other leading methods like SCI, JWNMF, and LJNMF, despite its enhanced functionality. This is particularly evident when comparing it to methods like LRL-GNMFT and LRL-GNMFA, which have an additional preprocessing complexity of $O(n^2d)$, while Div-JNMTF maintains a more efficient preprocessing step with $O(nlogn)$. This comparable complexity indicates that Div-JNMTF offers advanced functionality without sacrificing computational efficiency, making it a competitive choice for real-world applications.

Table 1
The computational complexity of the proposed model and the related methods.

Method	Complexity	
	Preprocessing	Overall
[16] SCI	–	$O(tn^2k + tdnk)$
[36] JWNMF	–	$O(tn^2k + tdnk)$
[34] CDE	$O(n\log n)$	$O(n\log n + tn^2k + tdnk)$
[35] LJNMF	–	$O(tn^2k + tdnk)$
[38] TANMF	–	$O(tn^2k + tdnk)$
[38] TASNMF	–	$O(tn^2k + tdnk)$
[41] NNRRNMF-ML	$O(n\log n)$	$O(n\log n + tn^2k + tdnk)$
[40] LRL-GNMFT	$O(n^2d)$	$O(n^2d + tn^2k + tdnk)$
[40] LRL-GNMFA	$O(n^2d)$	$O(n^2d + tn^2k + tdnk)$
Div-JNMTF	$O(n\log n)$	$O(n\log n + tn^2k + tdnk)$

4. Experimental results

In this section, we conduct extensive experiments to evaluate the effectiveness of the proposed Div-JNMTF approach in comparison to various attributed graph methods across four synthetic and eight real-world attributed networks. The following subsections present details on the datasets, evaluation metrics, compared methods, experimental results, and analysis.

4.1. Datasets

This paper evaluates the performance of the Div-JNMTF algorithm on four synthetic and eight widely used real attribute networks.

4.1.1. Synthetic datasets

We have utilized GenCAT [56] to generate synthetic attributed graphs for assessing the proposed method's performance. GenCAT enables the creation of graphs with customizable relationships between classes, attributes, and topology, allowing for fine-tuning via various parameters. This tool empowers users to control various aspects of graphs, including node characteristics (degree distribution, class membership), edge characteristics (class preferences), and attribute features (distribution and correlation). By adjusting parameters like class preference mean and class size distribution, users can simulate graphs with diverse properties such as homophily or heterophily. The parameters of GenCAT are configured as follows: c represents the number of classes, $\boldsymbol{M} \in \mathbb{R}^{c\times c}$ denotes the class preference matrix, defining the anticipated probability of connection between classes, $\boldsymbol{D} \in \mathbb{R}^{c\times c}$ signifies the class preference deviation matrix, indicating the variance in connection probability, and θ stands for the anticipated node degree. These parameters are variably set to generate various networks exhibiting diverse properties concerning class count, community structure, homophily, heterophily, and node attribute distributions. In this paper, to easily generate networks with different properties, we use the following formulas for constructing the class preference matrix $\boldsymbol{M}$ and the class preference deviation matrix $\boldsymbol{D}$:

$$\boldsymbol{M} = \delta\boldsymbol{I}_c + (1-\delta)\mathbf{1}_c, \quad \boldsymbol{D} = \sigma\mathbf{1}_c, \tag{33}$$

where δ and σ specify the cluster cohesion and cluster deviation, respectively, and $\mathbf{I}_c$ and $\mathbf{1}_c$ are the identity and all-ones matrices, respectively. More specifically, δ controls the strength of the community structure or homophily in the generated graph, while σ controls the degree of heterogeneity or variance in the connection probabilities between classes. By adjusting these parameters, GenCAT can generate synthetic graphs with different levels of homophily, heterophily, and community structure.

To evaluate robustness across dense and sparse networks, we have generated four attributed networks with different characteristics. The GenCat1Ke network is composed of 5 classes, generated with d = 100 attributes, n = 1024 nodes, mean θ = 4, cluster cohesion δ = 0.8, and cluster deviation σ = 0.2. This represents a moderately

Table 2
Details of the synthetic and real-world networks, where $|V|$: number of nodes, $|E|$: number of edges, θ: Mean Degree, d: number of attributes, c: number of clusters, δ: Cluster Cohesion, and σ: Cluster Deviation.

Dataset		$\|V\|$	$\|E\|$	θ	d	c	δ	σ
Synthetic	CatGen1Ke	1024	4218	4.12	100	5	0.8	0.2
	CatGen1Kh	1024	4233	4.13	100	5	0.7	0.3
	CatGen1Ks	1024	2227	2.17	100	5	0.8	0.2
	CatGen5K	5120	16503	3.22	500	10	0.8	0.2
Real-world	Texas	187	578	3.1	1703	5	–	–
	Cornell	195	569	2.92	1703	5	–	–
	Washington	230	783	3.4	1703	5	–	–
	Wisconsin	265	938	3.54	1703	5	–	–
	Wiki	2405	35 962	14.96	4973	17	–	–
	Cora	2708	5429	2	1433	7	–	–
	Citeseer	3312	4715	1.42	3703	6	–	–
	Blogcatalog	5196	343 486	66	8189	6	–	–

dense network with fairly cohesive clusters. Additionally, to generate a more challenging network with noise properties, we have generated the GenCat1Kh dataset, which shares similar parameters with d=100 attributes, n=1024 nodes, and $\theta = 4$, but with reduced cluster cohesion $\delta = 0.7$ and increased cluster deviation $\sigma = 0.3$. This introduces more noise and overlap between the clusters, making the representation task more difficult. To evaluate the proposed model on very sparse datasets, we have generated the GenCat1Ks, similar to GenCat1Ke but with a smaller $\theta = 2$. This sparse network puts the model to the test on graphs with fewer connections between nodes. Finally, to evaluate the ability of the proposed model on large-scale networks, the GenCat5K is generated, which has 5000 nodes, 500 attributes, and 10 clusters. This significantly larger network tests the scalability and performance of the approach on big data settings. The detailed statistics and characteristics of these synthetic networks are reported in Table 2 for reference.

4.1.2. Real-world datasets

In this paper, the effectiveness of the proposed model is evaluated on the eight real-world networks. These datasets include four webpage networks (Cornell, Texas, Washington, and Wisconsin) derived from WebKB [57], which represent the web page content of computer science departments in four US universities, incorporating hyperlink relationships and text content. Additionally, two citation networks (Cora and Citeseer) [58] are considered, comprising scientific publications from various fields, including references and content. The Wiki dataset [59], represents an encyclopedia network, consists of a compilation of Wikipedia documents that are inherently connected with each other via hyperlinks. Each document within this dataset is assigned to one or more predefined classes. BlogCatalog [60] is a site where bloggers may follow one another to establish a social media platform The eight networks contain topology and node attribute information. The basic information about these networks is presented in Table 2.

4.2. Evaluation metrics

To thoroughly evaluate the effectiveness of the proposed method, we employ three distinct evaluation metrics, Clustering Accuracy (ACC), Normalized Mutual Information (NMI), and Rand Index (RI). These metrics have different characteristics and collectively provide a comprehensive assessment of the clustering performance. The selection of metrics in this study is carefully considered based on their complementarity, widespread adoption, and proven effectiveness in evaluating clustering performance in unsupervised tasks and related works. A brief explanation of these metrics is provided in the following:

- **Clustering Accuracy (ACC)** is a commonly utilized metric in the fields of machine learning and data mining to assess the performance of various methods. ACC provides a measure of the overall correctness of a clustering or classification result. This measure is calculated as follows:

$$\mathrm{ACC}(\boldsymbol{y}, \boldsymbol{p}) = \frac{\sum_{i=1}^{n} \delta(y_i, map(p_i))}{n}, \tag{34}$$

In the given context, y_i represents the actual community assignment of the ith node, while p_i represents the community assignment detected by the algorithm for the same node. The mapping function $map(.)$ denotes the optimal mapping function used for comparison. The indicator function $\delta(.)$ is utilized, where its value is 1 if the values of its two parameters are equal and 0 otherwise. The ACC metric ranges from 0 to 1, indicating the degree of correct classification within a cluster.

- **Normalized Mutual Information (NMI)** is based on the concept of mutual information, which measures the amount of information that two sets of labels share. NMI normalizes the mutual information by the entropy of the two sets of labels, which makes it more suitable for comparing clustering results with different numbers of clusters. The formula for NMI is:

$$\mathrm{NMI}(\boldsymbol{y}, \boldsymbol{p}) = \frac{\mathrm{MI}(\boldsymbol{y}, \boldsymbol{p})}{\max(\mathcal{H}(\boldsymbol{y}), \mathcal{H}(\boldsymbol{p}))}. \tag{35}$$

where MI function measures the mutual information between two random variable, $\mathcal{H}$ function indicates the entropy of a random variable, and $\boldsymbol{y}$ and $\boldsymbol{p}$ represent the real community division and the community division result obtained in clustering method, respectively. NMI quantifies the similarity between two sets of labels, with 0 indicating no mutual information and 1 indicating perfect alignment.

- **Rand Index (RI)** is the pair counting-based metric of samples. The fundamental idea the RI is to evaluate how pairs of samples are clustered. This means that the quality of the identified communities is defined by measuring the fraction of concordant pairs of nodes between two partitions, $\boldsymbol{y}$ and $\boldsymbol{p}$. This definition can be expressed as follows:

$$\mathrm{RI}(\boldsymbol{y}, \boldsymbol{p}) = \frac{a + d}{a + b + c + d}, \tag{36}$$

Based on the text mentioned above, set a represents pairs of nodes that belong to the same communities in both partitions $\boldsymbol{y}$ and $\boldsymbol{p}$. Set b represents pairs of nodes that belong to the same communities in partition $\boldsymbol{y}$ but not in partition $\boldsymbol{p}$. Set c represents pairs of nodes that belong to the same communities in partition $\boldsymbol{p}$ but not in partition $\boldsymbol{y}$. Lastly, set d represents pairs of nodes that belong to different communities in both partitions $\boldsymbol{y}$ and $\boldsymbol{p}$. The RI measures the similarity between two clusterings, with 0 indicating randomness and 1 indicating perfect agreement with the ground truth.

4.3. Comparative algorithms

To assess the efficacy of the proposed Div-JNMTF method, it was compared against several well-known attributed graph clustering methods. These methods are as follows:

- **SCI** [16] is a method that extracts a community membership matrix from the topological structure and simultaneously attempts to map the attribute matrix to this extracted membership matrix.
- **CDE** [34] method partitions the structural information into subgraphs, and the node attributes are used to refine the community detection results.
- **LJNMF** [35] joins the symmetric NMF and basic NMF models using a complementary weighting strategy to fuse the topological and non-topological information.
- **TANMF** and **TASNMF** [38] are two parameter-free models that leverage topology information and attribute information. TANMF is based on the basic NMF model, while TASNMF utilizes the symmetric tri-factorization model. Both models integrate these two types of information without requiring additional parameters.

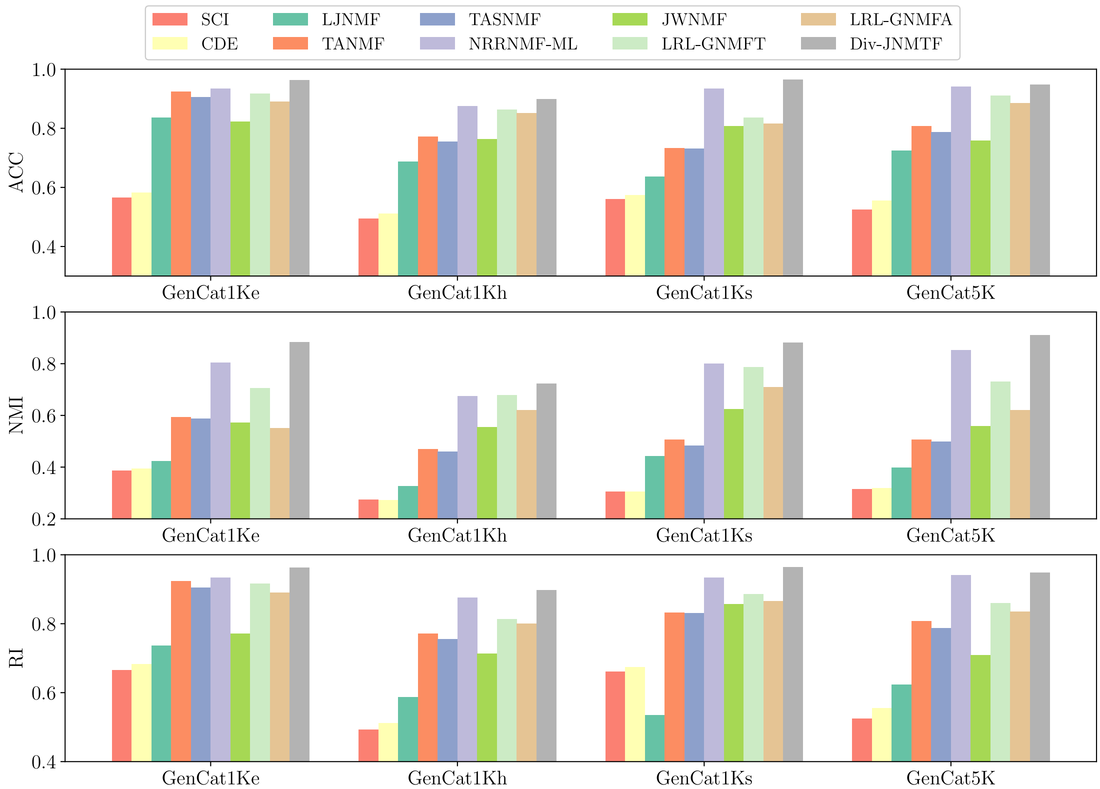


Fig. 2. The performance results on the synthetic attributed networks.

- **NRRNMF-ML** [41] is a manifold learning method for multi-view data clustering that combines basic NMFs with a non-redundant regularization.
- **JWNMF** [36] is a joint weighted NMF which combines Symmetric NMF and weighted NMF to integrate topology and attribute data.
- **LRL-GNMFT** and **LRL-GNMFA** [40] The LRL-GNMF method address two distinct scenarios. LRL-GNMFT focuses on nodes with limited attribute information, where the dominant view is based on node topology, and attribute information is treated as auxiliary information. On the other hand, LRL-GNMFA is designed for nodes with abundant attribute information, where the dominant view is based on attributes.

4.4. Clustering performance on synthetic datasets

We evaluated the performance of the proposed Div-JNMTF method along with several baseline and state-of-the-art models on four synthetic attributed datasets: GenCat1Ke, GenCat1Kh, GenCat1Ks, and GenCat5K. The clustering performance was assessed using ACC, NMI, and RI. From Fig. 2, we can see that Div-JNMTF appears to be highly competitive across all four GenCAT datasets and evaluation metrics. The provided bar graphs show Div-JNMTF consistently achieving higher or comparable scores, and it suggests strong performance and robustness. Performance is maintained across varying network densities and complexities. The results on GenCat1Ke, GenCat1Kh, GenCat1Ks, and GenCat5K demonstrate that Div-JNMTF can handle both dense and sparse networks, as well as larger-scale datasets with more classes and attributes. The proposed Div-JNMTF method achieved the highest scores across all three evaluation metrics on the four generated datasets. Div-JNMTF outperformed all other baseline and state-of-the-art methods evaluated, including the next best performer NRRNMFML. The basic models such as SCI and CDE exhibited relatively poor performance. Some other competitive methods displayed good results for certain metrics. In summary, the proposed Div-JNMTF method demonstrates robust clustering performance across diverse attributed networks, consistently achieving the highest scores on four synthetic datasets (GenCat1Ke, GenCat1Kh, GenCat1Ks, and GenCat5K) with varying complexities, densities, and scales. This method effectively handles both dense and sparse networks, showcasing scalability on the larger GenCat5K dataset. Div-JNMTF consistently outperforms baseline and state-of-the-art methods, including NRRNMFML, significantly surpassing basic models like SCI and CDE. Its superior performance is attributed to its unique combination of diversity regularization and efficient optimization algorithm, enabling effective clustering even in challenging network environments.

4.5. Clustering performance on real-world datasets

To assess the effectiveness of the Div-JNMTF model, we conducted a comparative analysis with nine established attributed community detection methods using eight distinct attributed network datasets. Tables 3–5 display the comparison results using ACC, NMI, and RI evaluation metrics. In these tables, the best results for each dataset are highlighted in bold, while the second-best results are underlined in the respective columns. The results demonstrate that the proposed method outperforms the other comparative methods across most datasets, as indicated by the considered evaluation metrics. These results validate the promising performance of the proposed approach in comparison to state-of-the-art methods. In few cases, the performance of the Div-JNMTF model slightly lags behind the other methods. For example, the TASNMF model achieves better performance on the Wisconsin dataset in terms of ACC and NMI. However, it is worth noting that the overall results of the proposed method significantly outperform the compared methods, which further emphasizes the efficiency of the Div-JNMTF model. When compared with the second-best methods, Div-JNMTF exhibits advantages in ACC by approximately 0.0672 and 0.0646 on the Cora and Texas datasets, improves NMI by around 0.0927 and 0.0787 on the Citeseer and Cora datasets, and surpasses other methods in RI by about 0.0656 and 0.0392 on the Texas and Cornell datasets. Overall, the proposed method outperforms the compared methods in 20 out

Table 3
The Clustering Accuracy (ACC) results on the real-world networks.

Method	Texas	Cornell	Washington	Wisconsin	Wiki	Cora	Citeseer	Blogcatalog
SCI	0.5896	0.4769	0.5435	0.5245	0.4187	0.4169	0.3442	0.3633
CDE	0.6150	0.6154	0.6696	0.7321	0.4357	0.3167	0.4927	0.3038
LJNMF	0.6498	0.5891	0.7253	0.7130	0.5320	0.4198	0.4808	0.3619
TANMF	0.6887	0.6237	0.7090	0.7480	0.4373	0.4163	0.4920	0.3281
TASNMF	0.6975	0.6254	0.7200	**0.7690**	0.3269	0.4061	0.4800	0.2941
NRRNMF-ML	0.6145	0.5892	0.6717	0.7053	0.2418	0.3796	0.2443	0.1893
JWNMF	0.5525	0.4307	0.4710	0.5120	0.2430	0.4130	0.2401	0.1967
LRL-GNMFT	0.6737	0.6617	0.6739	0.7433	**0.6045**	0.4239	0.5081	0.2028
LRL-GNMFA	0.6363	0.5589	0.6391	0.5849	0.5642	0.4128	0.4311	0.3052
Div-JNMTF	**0.7331**	**0.6774**	**0.7417**	0.7483	0.5390	**0.5026**	**0.6008**	**0.3786**

Table 4
The Normalized Mutual Information (NMI) results on the real-world networks.

Method	Texas	Cornell	Washington	Wisconsin	Wiki	Cora	Citeseer	Blogcatalog
SCI	0.2197	0.1520	0.2096	0.1852	0.2771	0.1780	0.0922	0.2213
CDE	0.3208	0.3403	0.4079	0.4284	0.2980	0.2117	0.2785	0.1846
LJNMF	0.3150	0.3047	0.3833	0.4302	0.4520	0.1484	0.2158	0.2316
TANMF	0.2985	0.3285	0.3659	0.4467	0.3344	0.1825	0.2066	0.2229
TASNMF	0.3039	0.3250	0.3758	**0.4611**	0.2942	0.1604	0.2060	0.2275
NRRNMF-ML	0.3222	0.2907	0.3755	0.4087	0.1302	0.1305	0.0164	0.0323
JWNMF	0.1335	0.0710	0.1593	0.0068	0.2129	0.1097	0.0151	0.0256
LRL-GNMFT	0.2877	0.3406	0.2985	0.4465	0.4607	0.1550	0.2368	0.0365
LRL-GNMFA	0.1921	0.1759	0.3297	0.1814	0.4498	0.1641	0.2174	0.1053
Div-JNMTF	**0.3868**	**0.3675**	**0.4377**	0.4486	**0.4632**	**0.2789**	**0.3167**	**0.2726**

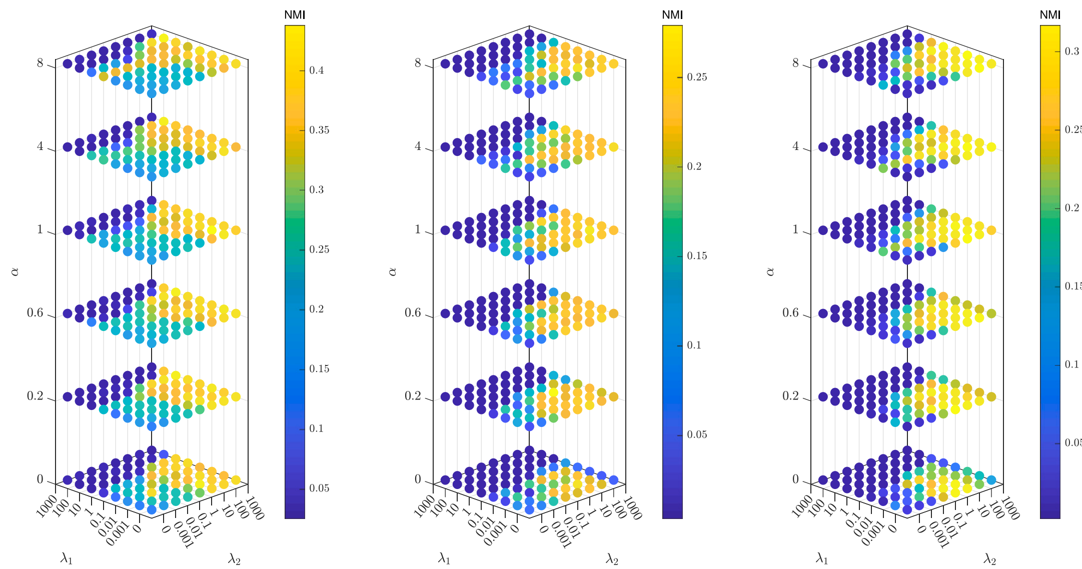
Table 5
The Rand Index (RI) results on the real-world networks.

Method	Texas	Cornell	Washington	Wisconsin	Wiki	Cora	Citeseer	Blogcatalog
SCI	0.6409	0.5727	0.5745	0.5615	0.7888	0.6345	0.5720	0.7125
CDE	0.5760	0.6115	0.6825	0.7290	0.7127	0.7363	0.7280	0.3810
LJNMF	0.5788	0.6162	0.7340	0.7394	0.8614	0.7519	0.7675	0.6127
TANMF	0.6457	0.6720	0.7208	0.7401	0.7072	0.7516	0.7644	0.5180
TASNMF	0.6290	0.6675	0.7327	0.7609	0.7044	0.7503	0.7687	0.5518
NRRNMF-ML	0.6190	0.6690	0.6590	0.6663	0.8437	0.6652	0.7121	0.2132
JWNMF	0.4633	0.4333	0.4358	0.4245	0.7760	0.6372	0.6075	0.5141
LRL-GNMFT	0.6407	0.7007	0.6819	**0.7714**	0.8649	0.7529	0.7730	0.2316
LRL-GNMFA	0.6368	0.6517	0.6917	0.6011	0.8638	0.7552	0.7571	0.5699
Div-JNMTF	**0.7113**	**0.7399**	**0.7486**	0.7403	**0.8674**	**0.7531**	**0.7907**	**0.7221**

of 24 cases, and it enhances the average results across all datasets by approximately 0.0338, 0.0318, and 0.0186 according to the ACC, NMI, and RI criteria, respectively.

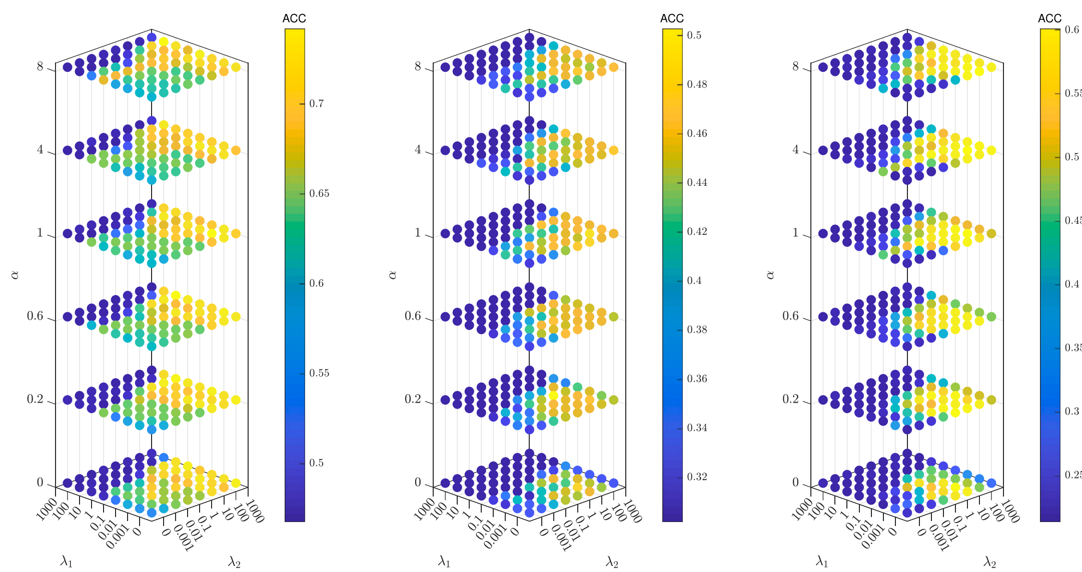
4.6. Parameter analysis

In this subsection, to investigate the impact of various parameters on the performance of the proposed method, a parameter analysis was conducted on the Washington, Cora, and Citeseer datasets. The proposed method involves three parameters: the attribute contribution parameter α, the HSIC parameter λ_1, and the local structure preservation parameter λ_2. In the experiments, the α parameter was adjusted within the range $\{0, 0.2, 0.6, 1, 4, 8\}$, while the λ_1 and λ_2 parameters were varied across $\{0, 0.001, 0.01, 0.1, 1, 10, 100, 1000\}$. The parameter sensitivity analysis process involved a grid search strategy, systematically evaluating all parameter combinations within the specified ranges for α, λ_1, and λ_2. To avoid overfitting and ensure robust parameter selection, we conducted the process with an average of 10 runs. This grid search process is computationally expensive; however, it allowed us to thoroughly explore the parameter space and identify the optimal parameter settings. The relationship between the α, λ_1, and λ_2 parameters and the performance of the proposed method are visualized in Figs. 3 and 4, representing the NMI and ACC measures, respectively. These figures are plotted in 3D, where each axis corresponds to α, λ_1, and λ_2, revealing an intricate pattern of performance variation as the parameters are adjusted. More specifically, to optimize model performance, it is preferable to set the λ_1 parameter to a small value below 10, prioritizing a modest setting. Excessive emphasis on HSIC regularization may overly prioritize independence between representations, potentially leading to the exclusion of vital information or the distortion of underlying data representations. Conversely, setting relatively large values for the λ_2 parameter tends to enhance performance. However, increasing λ_2 can enhance performance, yet excessively large values may cause network reconstruction terms to be dominated by manifold regularizations, resulting in decreased NMI and ACC values. These findings suggest that incorporating meaningful local structures into network reconstruction can be beneficial for the proposed network representation. Analysis from Figs. 3 and 4 indicates that tuning the α parameter is not overly sensitive, but excessively large values yield subpar performance. Nonetheless, for optimal NMI and ACC results, it is advisable to fine-tune the α parameters for each dataset, considering their specific characteristics, as different attribute networks offer varying levels of informative attributes. In summary, the analysis revealed that a modest value for λ_1 is preferable, while larger values for λ_2 tend to improve performance. Excessive values for either parameter, however, can negatively impact performance. The α parameter shows less sensitivity but tuning it for each dataset is still recommended to achieve optimal results. These results indicate the proposed model could combine structural information and attribute information efficiently, and further demonstrate the effectiveness of HSIC and dual



(a) Washington (b) Cora (c) Citeseer

Fig. 3. Parameter analysis of the proposed method in term of NMI measure.



(a) Washington (b) Cora (c) Citeseer

Fig. 4. Parameter analysis of the proposed method in term of ACC measure.

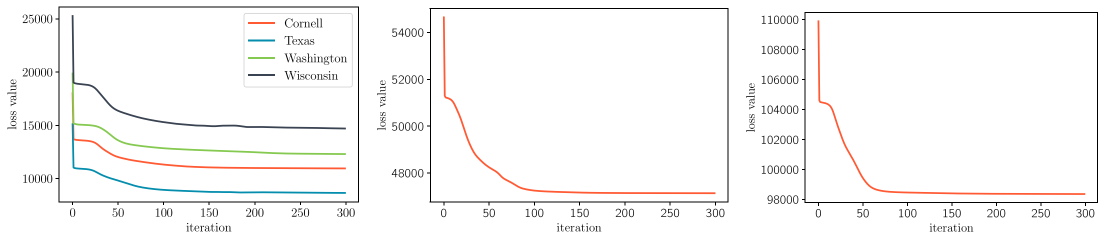
graph regularization terms in the attributed graphs clustering task. To clarify the parameter sensitivity, the optimal parameter values of the proposed model are depicted in Table 6, derived from the experimental findings.

4.7. Convergence analysis

In this subsection, we analyze the convergence of the proposed Div-JNMTF algorithm (Algorithm 1) through specific experiments conducted on seven networks. Fig. 5 displays the convergence results, where the x-axis represents the number of iterations and the y-axis represents the objective function value. The maximum number of iterations is set to 300 in the proposed method. To simplify the visualization, the WebKB datasets are shown in Fig. 5(a) due to their similar scales, while the other datasets are presented in separate subfigures. Clearly, by increasing number of iterations, the objective function values for all networks gradually decrease and eventually converges to a stable value. This experimental evidence confirms the complete convergence of the Div-JNMTF algorithm.

5. Conclusion

In this paper, we have presented the Diverse Joint Nonnegative Matrix Tri-Factorization for attributed graph clustering (Div-JNMTF), a novel model that combines two Nonnegative Matrix Tri-Factorization (NMTF) with distinct coefficients and shared correlations. By leveraging both the structural and attribute information of the graph, Div-JNMTF learns diverse representations that capture different aspects of the data. To address the issue of generating redundant features, we have utilized a Hilbert–Schmidt Independence Criterion (HSIC)



(a) WebKB (b) Cora (c) Citeseer

Fig. 5. Convergence analysis of the proposed Div-JNMTF model.

Table 6
The best parameter values of the Div-JNMTF model on the different datasets.

Dataset	Parameter		
	α	λ_1	λ_2
GenCat1Ke	4	0.1	100
GenCat1Kh	4	1	100
GenCat1Ks	4	0.01	10
GenCat5K	1	0.001	100
Texas	4	1	10
Cornell	0.2	1	100
Washington	4	0.001	100
Wisconsin	4	1	1000
Wiki	8	0.01	1000
Cora	0.6	0.1	10
Citeseer	8	0.001	100
Blogcatalog	0.1	0.001	1000

regularization term, which encourages the learned representations to be statistically independent. This regularization term promotes the extraction of non-redundant and informative features, leading to more meaningful clustering results. Furthermore, we have incorporated dual graph regularization terms into the model, which preserve the local structures of both topological and non-topological information. This ensures that the clustering algorithm captures the inherent characteristics of the attributed graph. The proposed Div-JNMTF integrates all these components into a unified objective function, enabling efficient optimization through the utilization of a multiplicative update rule. Experimental results on various datasets have demonstrated the superiority of the proposed model compared to existing methods. Div-JNMTF has consistently achieved better performance across three evaluation metrics.

There are several potential directions for future works that can further enhance the model and its applicability. Extending Div-JNMTF to handle dynamic graphs would enable the clustering of time-evolving attributed networks, capturing temporal dependencies and changes in the underlying structure. To capture the dynamic community structure, we can integrate both the network and its communities from the prior time step into the objective function. This allows for the simultaneous regularization of the topology and community arrangement from the previous time step. In addition, incorporating semi-supervised learning into the proposed Div-JNMTF model can be a valuable direction for future work. By leveraging a small set of labeled instances in conjunction with the available attribute and structural information, the model can potentially achieve even better clustering results. Finally, by integrating self-supervised learning techniques into the Div-JNMTF method, we can potentially boost learning from unlabeled data, enhancing the model's ability to capture diverse and informative representations in attributed graphs.

CRediT authorship contribution statement

Arina Mohammadi: Writing – original draft, Software, Methodology, Investigation, Data curation. **Seyed Amjad Seyedi:** Writing – original draft, Visualization, Software, Methodology, Conceptualization, Writing – review & editing. **Fardin Akhlaghian Tab:** Writing – review & editing, Supervision, Resources, Formal analysis, Conceptualization. **Rojiar Pir Mohammadiani:** Writing – review & editing, Validation, Project administration, Data curation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] I. McCulloh, H. Armstrong, A. Johnson, Social Network Analysis with Applications, John Wiley & Sons, 2013.
[2] S. Fortunato, Community detection in graphs, Phys. Rep. 486 (3) (2010) 75–174.
[3] M.E.J. Newman, Detecting community structure in networks, Eur. Phys. J. B 38 (2) (2004) 321–330.
[4] S. Wasserman, K. Faust, Social Network Analysis: Methods and Applications, Cambridge University Press, 1994.
[5] C. Bothorel, J.D. Cruz, M. Magnani, B. Micenková, Clustering attributed graphs: Models, measures and methods, Netw. Sci. 3 (3) (2015) 408–444.
[6] P. Chunaev, Community detection in node-attributed social networks: a survey, Comp. Sci. Rev. 37 (2020) 100286.
[7] S. Farzi, S. Kianian, A novel clustering algorithm for attributed graphs based on K-medoid algorithm, J. Exp. Theor. Artif. Intell. 30 (6) (2018) 795–809.
[8] Y. Li, C. Jia, X. Kong, L. Yang, J. Yu, Locally Weighted Fusion of Structural and Attribute Information in Graph Clustering, IEEE Trans. Cybern. 49 (1) (2017) 247–260.
[9] Z. Qu, J. Yang, X. Wang, S. Yin, Combining Link and Content for Community Detection in Social Networks, in: IEEE International Conference on Big Data and Smart Computing, BigComp, 2018, pp. 607–610.
[10] Z. Xu, Y. Ke, Y. Wang, H. Cheng, J. Cheng, A model-based approach to attributed graph clustering, in: Proceedings of the ACM SIGMOD International Conference on Management of Data, 2012, pp. 505–516.
[11] Y. Ruan, D. Fuhry, S. Parthasarathy, Efficient community detection in large networks using content and links, in: Proceedings of the 22nd International Conference on World Wide Web, 2013, pp. 1089–1098.
[12] D. He, Z. Feng, D. Jin, X. Wang, W. Zhang, Joint identification of network communities and semantics via integrative modeling of network topologies and node contents, in: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, 2017, pp. 116–124.
[13] L. Liu, L. Xu, Z. Wangy, E. Chen, Community Detection Based on Structure and Content: A Content Propagation Perspective, in: 2015 IEEE International Conference on Data Mining, 2015, pp. 271–280.
[14] Z. Bu, G. Gao, H.-J. Li, J. Cao, CAMAS: A cluster-aware multiagent system for attributed graph clustering, Inf. Fusion 37 (2017) 10–21.
[15] Z. Bu, H.-J. Li, J. Cao, Z. Wang, G. Gao, Dynamic cluster formation game for attributed graph clustering, IEEE Trans. Cybern. 49 (1) (2019) 328–341.
[16] X. Wang, D. Jin, X. Cao, L. Yang, W. Zhang, Semantic community identification in large attribute networks, in: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, 2016, pp. 265–271.
[17] M. Qin, D. Jin, D. He, B. Gabrys, K. Musial, Adaptive community detection incorporating topology and content in social networks, in: ASONAM '17: Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017, 2017, pp. 675–682.

[18] H. Chen, Y. Xiong, C. Wang, Y. Zhu, W. Wang, SpEC: Sparse embedding-based community detection in attributed graphs, in: Database Systems for Advanced Applications, 2020, pp. 53–69.
[19] Z. Huang, X. Zhong, Q. Wang, M. Gong, X. Ma, Detecting community in attributed networks by dynamically exploring node attributes and topological structure, Knowl.-Based Syst. 196 (2020) 105760.
[20] D.D. Lee, H.S. Seung, Learning the parts of objects by non-negative matrix factorization, Nature 401 (1999) 788–791.
[21] Y.-X. Wang, Y.-J. Zhang, Nonnegative matrix factorization: A comprehensive review, IEEE Trans. Knowl. Data Eng. 25 (6) (2012) 1336–1353.
[22] Z. Yang, Y. Xiang, K. Xie, Y. Lai, Adaptive method for nonsmooth nonnegative matrix factorization, IEEE Trans. Neural Netw. Learn. Syst. 28 (4) (2016) 948–960.
[23] D. Lee, H.S. Seung, Algorithms for non-negative matrix factorization, Adv. Neural Inf. Process. Syst. 13 (2000).
[24] A. Hajiveiseh, S.A. Seyedi, F. Akhlaghian Tab, Deep asymmetric nonnegative matrix factorization for graph clustering, Pattern Recognit. 148 (2024) 110179.
[25] S. Ghodsi, S.A. Seyedi, E. Ntoutsi, Towards cohesion-fairness harmony: Contrastive regularization in individual fair graph clustering, in: Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer, 2024, pp. 284–296.
[26] R. Abdollahi, S. Amjad Seyedi, M. Reza Noorimehr, Asymmetric semi-nonnegative matrix factorization for directed graph clustering, in: 2020 10th International Conference on Computer and Knowledge Engineering, ICCKE, 2020, pp. 323–328.
[27] U. von Luxburg, A tutorial on spectral clustering, Stat. Comput. 17 (4) (2007) 395–416.
[28] D. Kuang, C. Ding, H. Park, Symmetric nonnegative matrix factorization for graph clustering, in: Proceedings of the SIAM International Conference on Data Mining, SDM, 2012, pp. 106–117.
[29] F. Wang, T. Li, X. Wang, S. Zhu, C. Ding, Community discovery using nonnegative matrix factorization, Data Min. Knowl. Discov. 22 (3) (2011) 493–521.
[30] L. Lv, D. Bardou, Y. Liu, P. Hu, Deep Autoencoder-like non-negative matrix factorization with graph regularized for link prediction in dynamic networks, Appl. Soft Comput. 148 (2023) 110832.
[31] R. Mahmoodi, S.A. Seyedi, F. Akhlaghian Tab, A. Abdollahpouri, Link prediction by adversarial nonnegative matrix factorization, Knowl.-Based Syst. 280 (2023) 110998.
[32] R. Mahmoodi, S.A. Seyedi, A. Abdollahpouri, F. Akhlaghian Tab, Enhancing link prediction through adversarial training in deep Nonnegative Matrix Factorization, Eng. Appl. Artif. Intell. 133 (2024) 108641.
[33] Z. Shajarian, S.A. Seyedi, P. Moradi, A clustering-based matrix factorization method to improve the accuracy of recommendation systems, in: 2017 Iranian Conference on Electrical Engineering, ICEE, 2017, pp. 2241–2246.
[34] Y. Li, C. Sha, X. Huang, Y. Zhang, Community detection in attributed graphs: An embedding approach, in: AAAI Conference on Artificial Intelligence, Vol. 32, No. 1, 2018, pp. 338–345.
[35] Z. Chen, L. Li, H. Peng, Y. Liu, Y. Yang, Attributed community mining using joint general non-negative matrix factorization with graph Laplacian, Phys. A 495 (2018) 324–335.
[36] Z. Huang, Y. Ye, X. Li, F. Liu, H. Chen, Joint weighted nonnegative matrix factorization for mining attributed graphs, in: Advances in Knowledge Discovery and Data Mining, SpringerLink, 2017, pp. 368–380.
[37] S. Maekawa, K. Takeuch, M. Onizuka, Non-linear attributed graph clustering by symmetric NMF with PU learning, 2018, arXiv.
[38] D.-D. Lu, J. Qi, J. Yan, Z.-Y. Zhang, Community detection combining topology and attribute information, Knowl. Inf. Syst. 64 (2) (2022) 537–558.
[39] K. Berahmand, M. Mohammadi, F. Saberi-Movahed, Y. Li, Y. Xu, Graph regularized nonnegative matrix factorization for community detection in attributed networks, IEEE Trans. Netw. Sci. Eng. 10 (1) (2022) 372–385.
[40] R. Shang, W. Zhang, Z. Li, C. Wang, L. Jiao, Attribute community detection based on latent representation learning and graph regularized non-negative matrix factorization, Appl. Soft Comput. 133 (2023) 109932.
[41] G. Cui, Y. Li, Nonredundancy regularization based nonnegative matrix factorization with manifold learning for multiview data representation, Inf. Fusion 82 (2022) 86–98.
[42] A. Gretton, O. Bousquet, A. Smola, B. Schölkopf, Measuring statistical dependence with Hilbert-Schmidt norms, in: Algorithmic Learning Theory, SpringerLink, 2005, pp. 63–77.
[43] M. Jia, S. Liu, Y. Bai, Auto weighted robust dual graph nonnegative matrix factorization for multiview clustering, Appl. Soft Comput. 146 (2023) 110702.
[44] S.A. Seyedi, F. Akhlaghian Tab, A. Lotfi, N. Salahian, J. Chavoshinejad, Elastic adversarial deep nonnegative matrix factorization for matrix completion, Inform. Sci. 621 (2023) 562–579.
[45] S.A. Seyedi, P. Moradi, F.A. Tab, A weakly-supervised factorization method with dynamic graph embedding, in: 2017 Artificial Intelligence and Signal Processing Conference, AISP, 2017, pp. 213–218.
[46] M. Faraji, S.A. Seyedi, F. Akhlaghian Tab, R. Mahmoodi, Multi-label feature selection with global and local label correlation, Expert Syst. Appl. 246 (2024) 123198.
[47] M. Mozafari, S.A. Seyedi, R. Pir Mohammadiani, F. Akhlaghian Tab, Unsupervised feature selection using orthogonal encoder-decoder factorization, Inform. Sci. 663 (2024) 120277.
[48] S.A. Seyedi, S.S. Ghodsi, F. Akhlaghian, M. Jalili, P. Moradi, Self-paced multi-label learning with diversity, in: Proceedings of the Eleventh Asian Conference on Machine Learning, Vol. 101, 2019, pp. 790–805.
[49] C. Ding, T. Li, W. Peng, H. Park, Orthogonal nonnegative matrix T-factorizations for clustering, in: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2006, pp. 126–135.
[50] Y. Pei, N. Chakraborty, K. Sycara, Nonnegative matrix tri-factorization with graph regularization for community detection in social networks, in: Twenty-Fourth International Joint Conference on Artificial Intelligence, 2015, pp. 2083–2089.
[51] D. Cai, X. He, J. Han, T.S. Huang, Graph regularized nonnegative matrix factorization for data representation, IEEE Trans. Pattern Anal. Mach. Intell. 33 (8) (2011) 1548–1560.
[52] X. Li, H. Zhang, R. Zhang, Matrix completion via non-convex relaxation and adaptive correlation learning, IEEE Trans. Pattern Anal. Mach. Intell. 45 (2) (2023) 1981–1991.
[53] X. Zhang, A nonconvex relaxation approach to low-rank tensor completion, IEEE Trans. Neural Netw. Learn. Syst. 30 (6) (2019) 1659–1671.
[54] R. Xin, U.A. Khan, S. Kar, An improved convergence analysis for decentralized online stochastic non-convex optimization, IEEE Trans. Signal Process. 69 (2021) 1842–1858.
[55] J. Shermen, W. Morrison, Adjustment of an inverse matrix corresponding to changes in the elements of a given column or a given row of the original matrix, Annu. Math. Stat. 20 (1949) 621–625.
[56] S. Maekawa, Y. Sasaki, G. Fletcher, M. Onizuka, GenCAT: Generating attributed graphs with controlled relationships between classes, attributes, and topology, Inf. Syst. 115 (2023) 102195.
[57] W. Wang, X. Liu, P. Jiao, X. Chen, D. Jin, A unified weakly supervised framework for community detection and semantic matching, in: Advances in Knowledge Discovery and Data Mining, Springer, Cham, Switzerland, 2018, pp. 218–230.
[58] P. Sen, G. Namata, M. Bilgic, L. Getoor, B. Galligher, T. Eliassi-Rad, Collective classification in network data, AI Mag. 29 (3) (2008) 93.
[59] L. Cui, Z. Chen, J. Zhang, L. He, Y. Shi, P.S. Yu, Multi-view Collective Tensor Decomposition for Cross-modal Hashing, in: ICMR '18: Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval, Association for Computing Machinery, New York, NY, USA, 2018, pp. 73–81.
[60] W. Khan, M. Haroon, An unsupervised deep learning ensemble model for anomaly detection in static attributed social networks, Int. J. Cogn. Comput. Eng. 3 (2022) 153–160.