Learning Multi-level Topology Representation for Multi-view Clustering with Deep Non-negative Matrix Factorization

Zengfa Dou[a,*], Nian Peng[b,*], Weiming Hou[c], Xianghua Xie[d], Xiaoke Ma[b,**]

[a]*Schoool of Computer and Information Science, Qinghai Institute of Technology, Xining, Qinghai, China*
[b]*School of Computer Science and Technology, Xidian University, Xi'an, Shaanxi, China*
[c]*School of Information Science and Engineering, Hebei University of Science and Technology, Hebei, China*
[d]*Department of Computer Science, Swansea University, Swansea, United Kingdom*

Abstract

Clustering of multi-view data divides objects into groups by preserving structure of clusters in all views, requiring simultaneously takes into consideration diversity and consistency of various views, corresponding to the shared and specific components of various views. Current algorithms fail to fully characterize and balance diversity and consistency of various views, resulting in the undesirable performance. Here, a novel Multi-View Clustering with Deep non-negative matrix factorization and Multi-Level Representation (MVC-DMLR) learning is proposed, which integrates feature learning, multi-level topology representation, and clustering of multi-view data. Specifically, MVC-DMLR first learns multi-level representation (also called deep features) of objects with deep nonnegative matrix factorization (DNMF), facilitating the exploitation of hierarchical structure of multi-view data. Then, it learns multi-level graphs for each view from multi-level representation, where relations between diversity and consistency are addressed at various resolutions. MVC-DMLR integrates multi-level representation learning, multi-level topology representation learning and clustering, which is formulated as an optimization problem. Experimental results show the supe-

*These authors contribute equally
**Corresponding author
Email address: xkma@xidian.edu.cn (Xiaoke Ma)

riority of MVC-DMLR to baselines in terms of accuracy, F1-score, normalized mutual information and adjusted rand index.

Keywords: Multi-view clustering, Deep non-negative matrix factorization, Self-representation learning, multi-layer networks.

---

1. Introduction

As one of prevalent tasks in machine learning, clustering identifies groups of objects (clusters, modules, or communities) such that highly similar objects are assigned into the same groups, and dissimilar ones into different ones. The critical techniques of clustering are how to define similarity of objects, and how to perform assignment of objects. For example, K-means [1, 2] employs Euclidean distance as similarity, and performs assignment the nearest principle, i.e., each object is assigned to the nearest group. NMF [3] learns representation of objects in the low-dimensional subspace, where similarity of objects is exploited. And, spectral clustering [4, 5, 6] obtains closeness of objects in terms of spectral embedding that is associated with eigenvectors of matrices. However, the traditional algorithms target to clustering single-view data, i.e., data are observed from one perspective. Actually, one perspective is insufficient to fully characterize complex systems, resulting in multi-view data [7]. For example, movies attract audiences with multiple media, such as images, sounds and music, which maximize the stimulation of the audience's senses. In social networks, individuals communicate with counterparts with different manners, including emails, telephones and webchat.

Thus, it is of great significance to clustering of multi-view, which assigns objects into various groups such that highly similar objects in all views are assigned into the same groups, and those that are dissimilar for all views into different clusters. Multi-view clustering provides an insight into mechanisms of systems because comprehensive and accurate patterns in multi-view data [8]. Compared to clustering of single view data, multi-view clustering is much more complicated for two typical reasons. First, heterogeneity of various views

poses a great challenge, and how to address heterogeneity of various views is difficult. Second, clustering of multi-view data simultaneously takes into account similarity of objects within and across views, which are difficult to model and balance.

Luckily, many approaches are proposed with various strategies to address these challenges [8]. According to the principles of algorithms, available approaches are divided into four classes, i.e., co-training-, kernel-, graph-, and subspace-based ones, where methods in the first category apply single-view clustering to multi-view data with simply extension to avoid exploiting relations among various views. Actually, the simplest strategy is transformation, which converts multi-view data into one view, where clustering analysis is directly executed on collapsed data. However, it is criticized for performance of algorithms because intrinsic structure of data may be destroyed during transformation. To avoid destroying structure of data, co-training-based method [9, 10, 11] first performs clustering for a given view, and clusters are utilized to guide clustering of other views, which are criticized for the poor performance and robustness of algorithms. Specifically, performance of algorithms largely depends on the order of views for clustering, and these methods also fail to address heterogeneity of various views.

To attack this problem, the kernel-based methods [12, 13, 14, 15, 16] employ a kernel function function mapping objects onto a high dimensional space, where clustering is performed by integrating various views with a linear manner. The underlying assumption is that views are comparable in the kernel space, i.e., heterogeneity of multi-view data can be addressed with kernel functions. But, these algorithms are only inapplicable for multi-view data with complicated structure because they fail to maintain inherent structure of the original views, resulting in the low accuracy. Furthermore, selection of appropriate kernel functions for each view is also very complicated, which are determined by distribution and structure of views that are difficult to model. The graph-based methods [17, 18, 19, 20, 21] first construct a network for each view, and the perform clustering on the constructed networks by exploring topological struc-

ture of networks. These algorithms significantly improve the performance of clustering, demonstrating that graphs are promising in multi-view clustering field. Graph-based methods perfectly address heterogeneity of multi-view data, but they neglect relations among various views. And, the subspace-based methods [22, 23, 24, 25] projects all views into a shared subspace, where objects in various views are compatibly represented, facilitating the identification of consistence among various views. These algorithms dramatically enhance performance of multi-view clustering, proving that relations among various views are of great importance. However, selecting appropriate subspaces to represent various views is really difficult.

Although many excellent algorithms for multi-view clustering are developed, some critical and unsolved problems remain. Firstly, current algorithms concentrate on learning consistent representation of objects of various views in the shared algebra space, ignoring the intrinsic structure of representation. For example, the kernel-based methods focus on selecting of proper kernel functions that map objects onto kernel spaces, and subspace-based approaches are devoted to construct the low-rank shared subspace(s). Actually, objects of various views cannot be precisely represented with one type of space(s) because of the complexity of multi-view data. Second, available algorithms model consistency and diversity of various views by ignoring hierarchical structure of features, failing to fully characterize structure of clusters in multi-view data [26, 27, 18, 28]. Third, current algorithms separate feature learning, diversity and consistency learning, and clustering, failing to reach a tradeoff between these items. Recently, deep representation learning [29, 30, 31] is popular for the exploration of structure of complicated data, showing possibility of address relations of various views with deep features of objects.

To address problems mentioned above, a cohesive multi-view clustering with DNMF and multi-level representations (called MVC-DMLR) is proposed, which integrates feature learning, consistency and diversity learning, and clustering of multi-view data. MVC-DMLR is mainly composed of three components, i.e., deep feature learning, topology representation learning, and multi-view clus-

tering (Fig. 1). Specifically, MVC-DMLR employs DNMF to learn multi-level representation of objects, which explores the hierarchical structure of features of objects, improving capability of representation. To learn consistency and diversity of various views, MVC-DMLR decomposes multi-level topology representation into the consistency and diversity parts, which are characterized at various levels. In other words, relations of various views are characterized with different resolutions, significantly enhancing accuracy of relations of views. MVC-DMLR integrates all these procedures into an overall optimization problem. Experiments demonstrate that MVC-DMLR is better than baselines for clustering of multi-view data.

In all, contributions of this study are summarized as

- A novel multi-level topology representation is provided to exploit hierarchical structure of features, which explicitly quantifies consistency and diversity of various views, improving quality of features of objects.
- A novel integrative framework is proposed, which integrates deep feature learning, multi-level topology representation, and multi-view clustering. In this circumstance, representation is learned in accordance with clustering, which greatly improve performance of clustering.
- Experimental results prove superiority of MVC-DMLR to baselines, providing an effective model for multi-view clustering.

And, related work and preliminaries are summarized in Section 2 and 3, respectively. Section 4 and 5 present procedure and performance of MVC-DMLR, respectively. Section 6 concludes this study.

2. Related work

Clustering effectively analyzes large-scale data that cannot be manipulated as a whole, which divides objects into compact groups such that objects are similar inside of clusters, and dissimilar outside of clusters. And, multi-view clustering simultaneously takes into consideration similarity of objects within each view, and relations of across views. According to principles of algorithms,

current approaches belong to one of four typical classes, i.e., co-training-, kernel-, graph- and subspace-based approaches, where the first two categories focus on manipulating data, and the latter two ones concentrate on operating features of objects.

Co-training-based algorithms [9, 10, 11] assume that views are highly related, and clusters of one view is useful for guiding clustering of other views. Thus, these algorithms first perform clustering for a given view with the traditional single-view clustering, and then execute clustering of another view by setting clusters of the previous view(s) as prior, where relations among views are addressed with clusters. For example, Co-SC [9] and Co-Reg [10] first perform clustering for a given view, and then independently repeat clustering for the rest views by setting the previous obtained clusters as prior information. In essence, co-training-based algorithms simply extend single-view clustering and apply to multi-view scenarios with a sequential manner. These algorithms are simple and easy to understand, which are unpopular for the low accuracy. First, performance of algorithms is instable because orders of views for clustering dramatically effect performance of clustering because errors of clusters in one view are very likely be accumulated in other views. Second, relations among views are degenerated to relation between a pair of views, failing to fully address correlation among them.

To fully address relations among views, the kernel-based algorithms [12, 13, 14, 15, 16] hypothesize that there exists an space, where all views are compatible so that the relations among views can be directly modeled. Therefore, these algorithms employ a kernel function to map objects in each view onto a high dimensional kernel space, and clustering is performed with a linear manner to combine all these views. For instance, KSCC [12] employs non-flat manifolds to hybrid linear model for the construction of kernel functions, and Tzortzis *et al.* [13] assume that weights for kernel functions are related to quality of views. Generally speaking, kernel-based algorithms promote performance of co-training approaches because relations of views are addressed. However, these algorithms are unsatisfied because selecting appropriate kernels without deeply

Table 1: Main symbols and description.

Symbol	Definition and description
$\mathcal{X}$	Multi-view data $\{X^{[1]}, \ldots, X^{[\nu]}\}$
$X^{[v]}$	Profile of the v-th view
m_v	Number of attributes in $X^{[v]}$.
$B^{[v,i]}$	The i-th level basis matrix of deep NMF for $X^{[v]}$
$F^{[v,i]}$	The i-th level coefficient matrix of deep NMF for $X^{[v]}$
$C^{[v]}$	The conserved topology representation for $X^{[v]}$
$U^{[v,i]}$	The i-th level-specific topology representation for $X^{[v]}$
$Tr(X)$	Trace of X i.e. $tr(X) = \sum_i x_{ii}$
W	Adjacent matrix of affinity graph G

understanding of properties of multi-view data is difficult.

However, these approaches directly manipulate data for clustering, where the high-order relations among objects are ignored. To avoid this problem, graph-based methods [17, 18, 19, 20, 21] construct graph(s) for various views, transforming the original problem into graph clustering. Thus, clusters in multi-view data is equivalent to communities in network(s). The great difference among these algorithms lies on the construction and clustering of graph(s). For instance, GMC [19] constructs an unified graph of various views with the mutual reinforcement learning, where clusters are directly obtained from the unified graph. CGDD [20] simultaneously and explicitly exploits consistency and cross-graph diversity of various views, where noise of each view is effectively removed. Graph-based algorithms dramatically improve performance of clustering of multi-view data, demonstrating that topological structure of graphs of views is critical for multi-view clustering.

Furthermore, subspace-based algorithms learns the low-dimensional representation of objects by projecting all views into subspace(s), where the key technique is to find appropriate subspace(s) [32]. For example, Convex [32] projects all views into a shared subspace with convex optimization, whereas MVSC [33] performs subspace clustering by preserving the consistence of clusters among different views. To further exploit hierarchical representation of multi-view data, DNMF [34] learn deep features of objects to obtain the multi-level representa-

tion, where hierarchical structure of features is investigated, thereby providing a more comprehensive way to model multi-view data. DMVC [35] utilizes deep features of objects for clustering. awDMVC [11] automatically learns weights for various views, whereas MVC-DMF-PA [36] employs alignment of partitioning of various views for clustering. In addition, DANMF-MRL [31] explores consistency of views with deep features of objects. Nevertheless, these algorithms only concentrate on the identification of the shared features, ignoring view-specific features, which is also encouraging for multi-view clustering. For instance, DiMSC [28] quantifies diversity of features of objects to obtain the complementary information with Hilbert-Schmidt independence criterion, and FMR [37] explores diversity of features of various views.

Here, we investigate possibility of exploiting consistency and diversity of multi-level representation across views to fully characterize multi-view data.

3. Preliminaries

In this study, letters denote variables, where uppercase, lowercase boldface and lowercase letters represent matrices, vectors and scalars, respectively. Matrix $X \in R^{m\times n}$ is profile of n objects with m features, where element x_{ij} denotes value of the i-th feature of the j-th object. Multi-view data is denoted by $\mathcal{X} = \{X^{[1]}, \ldots, X^{[\nu]}\}$, where $X^{[v]} \in R^{m_v\times n}$ is the profile of the v-th view, and ν is the number of views. $\|X\| = \sqrt{\sum_{ij} x_{ij}^2}$ is the Frobenious norm of X, and $X^{'}$ is the transpose of matrix X. Let $\mathbf{x}_{i.}(X_{i.})$ and $\mathbf{x}_{.j}(X_{.j})$ be the i-th row and j-th column of matrix X. If X is a square matrix, $Tr(X)=\sum_i x_{ii}$ is trace of X. Given feature matrix X, a graph $G = (V, E)$ is constructed (V and E are the set of vertices and edges respectively), whose adjacent matrix is denoted by $W \in R^{n\times n}$ with element w_{ij} as weight on edge connecting the i-th and j-th vertex. Graph clustering divides vertex set into groups such that vertices within the same groups are well connected, and weakly connected across different clusters. The major notations are summarized in Table 1.

NMF [3] approximates data X with two low-dimensional and nonnegative

matrices such that minimization of error is achieved, i.e.,

$$\min_{B,F} \|X - BF\|^2, \quad s.t. \quad B \geq 0, F \geq 0, \tag{1}$$

where $B \in R^{m \times k}$ and $F^{k \times n}$ are the basis and coefficient matrix, respectively. DNMF [38, 34] learns hierarchical representation of the original data X as

$$\min_{B^{[i]},F} \|X - B^{[1]}B^{[2]} \cdots B^{[\tau]}F\|^2, \quad s.t. \quad B^{[i]} \geq 0, F \geq 0, \tag{2}$$

where $B^{[i]}$ and F denote the i-th basis and coefficient matrix, and τ is the number of levels. Notice that τ=1 implies that DNMF is degenerated to NMF.

Self-representation learning [39] hypothesizes that objects can be represented with others with a linear function, i.e.,

$$\mathbf{x}_i = \sum_{j,j \neq i} \psi_j \mathbf{x}_j, \tag{3}$$

where ψ_j is the weight for the j-th object. Graph can be learned from self-representation is as

$$\min \|X - X\Psi\|^2, \quad s.t. \quad diag(\Psi) = 0, \tag{4}$$

where diagonal elements of Ψ are 0 to avoid trivial solutions.

4. The proposed algorithm

In this section, the model, optimization, and analysis of MVC-DMLR are presented.

4.1. Objective function

The objective function of MVC-DMLR is composed of three major components, where each of them corresponds to one of procedures, i.e., deep feature learning, multi-level topology representation learning and clustering as shown in Fig. 1.

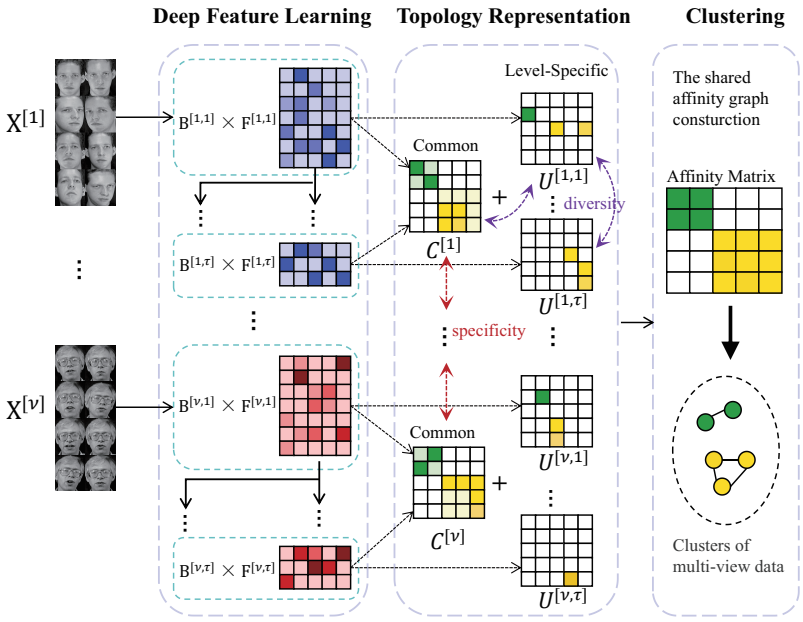

Figure 1: Overview of the MVC-DMLR algorithm, which consists of deep feature learning, multi-level topology learning and clustering, where the first procedure learns multi-level representation with DNMF, multi-level topology representation learning measures consistency and diversity of various views by manipulating multi-level features of objects, and clustering procedure identifies clusters, respectively.

On the deep feature learning issue, the most frequently employed approach is to obtain low-dimensional features of objects with NMF [3]. Specifically, NMF approximates profile $X^{[v]}$ of each view with low-rank nonnegative matrices by minimizing reconstruction error, i.e.,

$$\|X^{[v]} - B^{[v]}F^{[v]}\|^2, \quad s.t. \quad B^{[v]} \geq 0, F^{[v]} \geq 0. \tag{5}$$

To exploit relations of various views, features of objects can be obtained by summing all views as

$$\sum_{v=1}^{\nu} \|X^{[v]} - B^{[v]}F^{[v]}\|^2. \tag{6}$$

However, Eq.(6) fails to discriminate intrinsic complicated structure of features, since objects in multi-view data cannot be precisely characterized with features from one level. Therefore, there is a critical need for learning multi-level representation of objects, where each level characterizes structure at various resolution, thereby enhancing comprehensibility of features. DNMF [34] in Eq.(2) learns multi-level representation of objects, where $B^{[i]} \cdots B^{[\tau]}F$ denotes the i-th level representation ($1 \leq i \leq \tau$). By jointly factorizing profiles of views, Eq.(6) is reformulated as

$$\mathcal{L}_{DNMF} = \sum_{v=1}^{\nu} \|X^{[v]} - B^{[v,1]} \cdots B^{[v,\tau]}F^{[v,\tau]}\|^2, \tag{7}$$

where $F^{[v,\tau]}$ feature of objects. In this case, hierarchical structure of features is learned, which is more precise to model complicated structure of multi-view data. In other words, feature $F^{[v,i]} = B^{[v,i+1]} \cdots B^{[v,\tau]}F^{[v,\tau]}$ describes the global structure if i is small, local details otherwise. Furthermore, multi-level representation also provides an opportunity to investigate intrinsic and complicated structure of multi-view at different resolutions, thereby improving the quality of features.

On multi-level topology representation learning issue, current algorithms [34, 36] only makes use of the last level feature $F^{[v,\tau]}$ to perform clustering,

whereas it is insufficient to fully characterize multi-view data since it merely describes details of data, ignoring meta-structure of objects. Moreover, relations among various views is also lacking, and MVC-DMLR employs self-representation learning in Eq.(**??** to learn an affinity graph for each level as

$$\|F^{[v,i]} - F^{[v,i]}W^{[v,i]}\|^2, \quad s.t. \quad diag(W^{[v,i]}) = 0, \tag{8}$$

where constraint $diag(W^{[v,i]})$=0 requires diagonal elements are 0 to avoids trivial solutions, i.e., samples are expressed by themselves. By summing all levels, Eq.(8) is re-written as

$$\sum_{i=1}^{\tau} \|F^{[v,i]} - F^{[v,i]}W^{[v,i]}\|^2. \tag{9}$$

By jointly learning all views, MVC-DMLR obtains multi-level topology representation for multi-view data as

$$\mathcal{L}_{SR} = \sum_{v=1}^{\nu}\sum_{i=1}^{\tau} \|F^{[v,i]} - F^{[v,i]}W^{[v,i]}\|^2. \tag{10}$$

Eq.(10) results in consequent advantage that hierarchical structure is explicitly learning, thereby facilitating the quantification of consistency and diversity of various views.

However, directly clustering of the constructed graphs is impractical because of two reasons. First, our previous studies [40, 41, 42] demonstrate that it is complicated to directly exploit structure of multi-layer networks. Second, diversity and consistency of various graphs are ignored, hampering the identification of clusters. To address this problem, MVC-DMLR automatically separates $W^{[v,i]}$ into the conserved and level-specific parts, where the former part reflects consistency of all levels, and the latter one characterizes diversity of each view. In details, MVC-DMLR decomposes $W^{[v,i]}$ into the conserved part $C^{[v]}$ and

level-specific part $U^{[v,i]}$ such that

$$W^{[v,i]} = C^{[v]} + U^{[v,i]}. \tag{11}$$

By substituting the above expression into Eq.(10), we reformulate it as

$$\mathcal{L}_{SR} = \sum_{v,i} \|F^{[v,i]} - F^{[v,i]}(C^{[v]} + U^{[v,i]})\|^2. \tag{12}$$

Furthermore, l_2 norm [43] is imposed onto level-specific part as

$$\mathcal{L}_{REG} = \sum_{v,i} \|U^{[v,i]}\|^2. \tag{13}$$

The relations among consistency C and diversity U are further addressed, where orthogonality [44] is employ to ensure separation of them as

$$C^{[v]}(U^{[v,i]})' = I, \tag{14}$$

where I is the identity matrix. And, it can be formulated as trace minimization, i.e.,

$$Tr(C^{[v]}(U^{[v,i]})'). \tag{15}$$

Furthermore, we also expect diversity across various views is also addressed with orthogonality as

$$\begin{aligned} \mathcal{L}_{DIV} = &\sum_{v_p \neq v_q} Tr(C^{[v_p]}(C^{[v_q]})') + \sum_{v,i} Tr(C^{[v]}(U^{[v,i]})') \\ &+ \sum_{v} \sum_{i_j \neq i_k} Tr(U^{[v,i_j]}(U^{[v,i_k]})') \end{aligned} \tag{16}$$

By combining Eq.(7), (12), (13), and (16), the objective function of MVC-

DMLR is finalized as

$$
\begin{aligned}
\mathcal{L} &= \mathcal{L}_{DNMF} + \mathcal{L}_{SR} + \lambda\mathcal{L}_{REG} + \gamma\mathcal{L}_{DIV} \\
&= \sum_{v=1}^{\nu} \|X^{[v]} - B^{[v,1]} \cdots B^{[v,\tau]} F^{[v,\tau]}\|^2 \\
&+ \sum_{v,i} \|F^{[v,i]} - F^{[v,i]}(C^{[v]} + U^{[v,i]})\|^2 + \lambda \sum_{v,i} \|U^{[v,i]}\|^2 \\
&+ \gamma \sum_{v_p \neq v_q} Tr(C^{[v_p]}(C^{[v_q]})') + \gamma \sum_{v,i} Tr(C^{[v]}(U^{[v,i]})') \\
&+ \gamma \sum_{v} \sum_{i_j \neq i_k} Tr(U^{[v,i_j]}(U^{v,i_k]})') \\
s.t. \quad & B^{[v,i]} \geq 0, F^{[v,i]} \geq 0, C^{[v]} \geq 0, U^{[u,i]} \geq 0, \\
& diag(C^{[v]}) = 0,\ diag(U^{[v,i]}) = 0
\end{aligned}
\tag{17}
$$

where parameter λ and γ control importance of diversity and its relations with consistency, respectively.

On clustering of multi-view data issue, MVC-DMLR constructs an affinity graph from various views as

$$
W = \frac{1}{\nu} \sum_{v} \frac{C^{[v]} + (C^{[v]})'}{2} + \frac{1}{\nu\tau} \sum_{v,i} \frac{U^{[v,i]} + (U^{[v,i]})'}{2}. \tag{18}
$$

And, we subsequently employ spectral clustering to obtain clusters with affinity graph W.

4.2. Optimization

We adopt ADMM [45] to optimize Eq.(17), which alternatively updates a variable by fixing others.

Optimization of $B^{[v,i]}$: By removing irrelative items, the objective function in terms of $B^{[v,i]}$ is reformulated as

$$
\mathcal{L} = \|F^{[v,i-1]} - B^{[v,i]} F^{[v,i]}\|^2 \quad s.t. \quad B^{[v,i]} \geq 0, \tag{19}
$$

where

$$F^{[v,i-1]} = \begin{cases} X^{[v]}, & \text{if} \quad \text{i= 1}, \\ F^{[v,i-1]}, & \text{otherwise.} \end{cases} \tag{20}$$

And, Eq.(19) is standard NMF, which can be effectively solved with the adding and multiplication strategy as [46]

$$B^{[v,i]} = B^{[v,i]} \odot \frac{F^{[v,i-1]}(F^{[v,i]})'}{B^{[v,i]}F^{[v,i]}(F^{[v,i]})'}, \tag{21}$$

where $\odot$ denotes the Hadamard product.

Optimization of $F^{[v,i]}$*:* Eq.(17) is equivalent with the following problem

$$\mathcal{L} = \|F^{[v,i-1]} - B^{[v,i]}F^{[v,i]}\|^2 \quad s.t. \quad F^{[v,i]} \geq 0 \tag{22}$$

where

$$F^{[v,i-1]} = \begin{cases} X^{[v]}, & i\text{=1}, \\ F^{[v,i-1]}, & \text{otherwise.} \end{cases} \tag{23}$$

Notice that Eq.(22) is convex, where the analytical solution exists. the partial derivative of $\mathcal{L}$ in terms of $F^{[v,i]}$ is formulated as

$$\frac{\partial \mathcal{L}}{\partial F^{[v,i]}} = (B^{[v,i]})'(F^{[v,i-1]} - B^{[v,i]}F^{[v,i]}) \tag{24}$$

By setting partial derivative $\frac{\partial L}{\partial F^{[v,i]}}$=0, the update rule is obtained as

$$F^{[v,i]} = F^{[v,i]} \odot \frac{(B^{[v,i]})'F^{[v,i-1]}}{(B^{[v,i]})'B^{[v,i]}F^{[v,i]}}. \tag{25}$$

Optimization of $C^{[v]}$: By removing irrelevant item for $C^{[v]}$, objective function

is transformed into

$$\begin{aligned}\mathcal{L} =& \sum_{v,i} \|F^{[v,i]} - F^{[v,i]}(C^{[v]} + U^{[v,i]})\|^2 \\ &+\gamma \sum_{v_p \neq v_q} Tr(C^{[v_p]}(C^{[v_q]})^{'}) + \gamma \sum_{v,i} Tr(C^{[v]}(U^{[v,i]})^{'}) \\ &s.t. \quad diag(C^{[v]}) = 0, C^{[v]} \geq 0.\end{aligned} \tag{26}$$

The partial derivative of Eq.(26) in terms of $C^{[v]}$ is deduced as

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial C^{[v]}} =& \sum_i (F^{[v,i]})^{'} F^{[v,i]} C^{[v]} - \sum_i (F^{[v,i]})^{'} F^{[v,i]} \\ &+ \sum_i (F^{[v,i]})^{'} F^{[v,i]} U^{[v,i]} + \gamma (\sum_{v_p \neq v} C^{[v_p]} + \sum_i U^{[v,i]}).\end{aligned} \tag{27}$$

By setting $\frac{\partial \mathcal{L}}{\partial C^{[v]}}$=0, the update rule for $C^{[v]}$ is obtained as

$$C^{[v]} = C^{[v]} \odot \frac{\sum_i (F^{[v,i]})^{'} F^{[v,i]}}{\phi(F^{[v,i]}, C^{[v]}, U^{[v,i]})}, \tag{28}$$

where $\phi(F^{[v,i]}, C^{[v]}, U^{[v,i]}) = \sum_i (F^{[v,i]})^{'} F^{[v,i]}(C^{[v]} + U^{[v,i]}) + \gamma(\sum_{v_p \neq v} C^{[v_p]} + \sum_i U^{[v,i]})$.

Optimization of $U^{[v,i]}$: Eq.(17) in terms of $U^{[v,i]}$ is equivalent with the following problem

$$\begin{aligned}\mathcal{L} =& \sum_{v,i} \|F^{[v,i]} - F^{[v,i]}(C^{[v]} + U^{[v,i]})\|^2 + \lambda \sum_{v,i} \|U^{[v,i]}\|^2 \\ &+ \gamma \sum_v \sum_{i_j \neq i_k} Tr(U^{[v,i_j]}(U^{v,i_k]})^{'}) + \gamma \sum_{v,i} tr((C^{[v]})^{'} U^{[v,i]}) \\ &s.t. \quad diag(U^{[v,i]}) = 0, U^{[v,i]} \geq 0.\end{aligned} \tag{29}$$

And, the partial derivative of $\mathcal{L}$ in terms of $U^{[v,i]}$ is deduced as

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial U^{[v,i]}} &= (F^{[v,i]})' F^{[v,i]} U^{[v,i]} - (F^{[v,i]})' F^{[v,i]} \\ &+ (F^{[v,i]})' F^{[v,i]} C^{[v]} + \lambda U^{[v,i]} \\ &+ \gamma \sum_{i_j \neq i} U^{[v,i_j]} + \gamma C^{[v]}.\end{aligned} \tag{30}$$

By setting the partial derivative $\frac{\partial \mathcal{L}}{\partial U^{[v,i]}}$ as 0, the update rule for $U^{[v,i]}$ is formulated as

$$U^{[v,i]} = U^{[v,i]} \odot \frac{(F^{[v,i]})' F^{[v,i]}}{\varphi(F^{[v,i]}, C^{[v]}, U^{[v,i]})}, \tag{31}$$

where $\varphi(F^{[v,i]}, C^{[v]}, U^{[v,i]}) = (F^{[v,i]})' F^{[v,i]}(C^{[v]} + U^{[v,i]}) + \lambda U^{[v,i]} + \gamma(\sum_{i_j \neq i} U^{[v,i_j]} + C^{[v]})$.

Algorithm 1 MVC-DMLR Algorithm

Input:

$\mathcal{X}$: Multi-view data;

λ, γ : Regularization parameters;

Output:

Clusters of $\mathcal{X}$;

Part I: Deep feature learning

1. Initialize matrix $B^{[v,i]}$ and $F^{[v,i]}$ with Singular Value decomposition as Ref.[47];
2. Update matrix $B^{[v,i]}$ according to Eq.(21);
3. Update matrix $F^{[v,i]}$ according to Eq.(25);

Part II: Multi-level topology representation

4. Update matrix $C^{[v]}$ according to Eq.(28);
5. Update matrix $U^{[v,i]}$ according to Eq.(31);
6. Go to step 2 until convergence;

Part III: Clustering

7. Construct the affinity matrix W as Eq. (18).
8. Clustering W using spectral clustering.

4.3. Algorithm analysis

On the space complexity of MVC-DMLR, it needs space $O(n\sum_v m_v)$ to store $\{X^{(v)}\}_{v=1}^{\tau}$. For the deep feature learning, space for the basis and coefficient matrices is $O(n\sum_v m_v)$ since NMF is dimension reduction-based method. Moreover, MVC-DMLR takes space $O(n^2\tau)$ to store graphs, and space for graph-

s in Eq.(18) is $O(n^2)$. Therefore, the total space complexity of MVC-DMLR is $O(n^2\tau)$.

On the time complexity of MVC-DMLR, it updates matrix $B^{[v,i]}$, $F^{[v,i]}$, $C^{[v]}$ and $U^{[v,i]}$. And, time for updating matrices is $O(n^2lk)$, where l is the number of iterations, and k is the number of features. Thus, the time complexity of MVC-DMLR is $O(lkn^2\nu\tau))$. Even though time complexity of MVC-DMLR is higher than that of NMF, we demonstrate that MVC-DMLR is also efficient (Section 5.4).

Table 2: Statistics of multi-view data, where n, ν and m_v denote the number of objects, views, and attributes, respectively.

	n	ν	m_v
BBC	685	4	4659/4633/4665/4684
BBCSport	544	2	3183/3203
3sources	169	3	3560/3631/3068
Reuters	1200	5	2000/2000/2000/2000/2000
CiteSeer	3312	2	3312/3703
WebKB	203	3	1703/230/230
NGS	500	3	2000/2000/2000
ORL	400	3	4096/3304/6750
Yale	165	3	4096/3304/6750
100leaves	1600	3	64/64/64

5. Experimental results

Extensive experiments are conducted to testify effectiveness of MVC-DMLR with seven state-of-the-art baselines and ten benchmark datasets.

5.1. Data

Ten datasets are chosen for experiments, which are depicted as

- **BBC** [48] involves 685 documents from BBC news, and each of them with 4 views. The number of attributes of views are 4659, 4633, 4665 and 4684, respectively.
- **BBCSport** [48] involves 544 documents from BBC Sport column, and each document is depicted into two views with 3183 and 3203 attributes, respectively.

- **3Sources** consists of 169 news from 3 news organizations, where samples are manually labeled as one of seven topical labels.
- **Reuters** [49] contains 1200 documents, and each document is written by using five languages, such as Englishi, Italian, French, German, and Spanish, where each language has 2000 words as features.
- **CiteSeer** [49] contains 3312 documents over 6 labels, and each one is associated with the content and citations view, where the content view has 3703 words, and citation view includes 4732 links, respectively.
- **WebKB** is composed of 203 web pages, which is gathered from the computer science department of University of California. Each sample is depicted by the materials, recording the anchor text on the hyperlink, and titles as attributes.
- **20NGs** is a collection of news documents drawn from 20 distinct newsgroups, each of which has 500 instances that are processed by 3 various approaches.
- **ORL** is from the Olivetti Research Laboratory in Cambridge, which consists of 40 distinct objects, and each of them is with 10 diverse images. Three different attributes are used to depicted an image including 4096 intensity, 3304 LBP, and 6750 Cabor, respectively.
- **Yale** [50] consists of 165 raw images belonging to 15 subjects, and there are 11 images under various environments for each subject. For each image, three types of features are generated.
- **100leaves** consists of 1600 samples from 100 plant species, where features include shape descriptor, fine-scale margin, and texture histogram.

All these datasets can be categorized into two classes, i.e., texts and images, where BBC, BBCSport, 3Sources, Reuters, CiteSeer, WebKB, and 20NGs belong to the first class, and ORL, Yale, and 100leaves are the second one. The statistics of these datasets are summarized in Table 2.

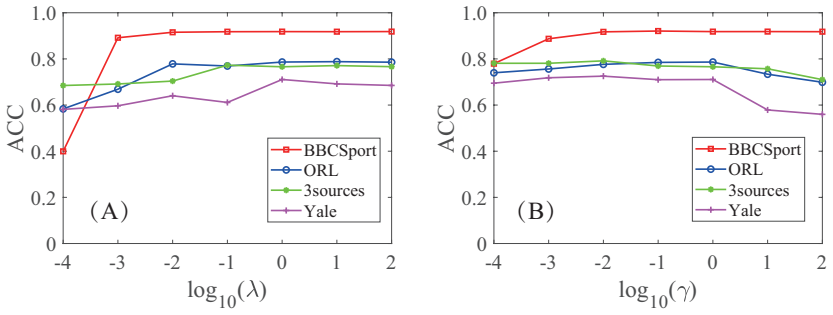

Figure 2: ACC of MVC-DMLR vs various parameters: (A) log of λ, and (B) log of γ.

5.2. Baselines and metrics

Seven state-of-the-art baselines are chosen for a comparison to fully validate performance of the proposed algorithm, including as Co-Reg [10], MultiNMF [51], DMVC [35], CSMSC [22], GMC [19], MVC-DMF-PA [36], and CGDD [20], which cover all typical multi-view clustering approaches. All these algorithms are executed on HP Z228 workstation with Intel i5 CPU, 64G memory and 1Tb RAM with the optimal values of parameters. MVC-DMLR is coded with python.

To evaluate the clustering performance, four popular metrics, such as accuracy (ACC), normalized mutual information (NMI) [52], F-score, and adjusted rand index (ARI) [53], are selected as evaluation criteria. Each algorithm is executed 50 time, and the mean of accuracy is selected to measure performance of algorithms. ACC is defined as

$$ACC = \frac{1}{n}\sum_{i=1}^{n}\delta(p_i, g_i), \tag{32}$$

where p_i and g_i denote the predicted and truth label of the i-th object respectively, and $\delta(p_i, g_i)$ is an indicator function that is 1 if $p_i = g_i$, 0 otherwise.

Let C^* and C be the ground truth and predicted clusters, NMI [52] constructs a confusion matrix P with the element p_{ij} as the number of vertices overlapped by the i-th true cluster in C^* and the j-th predicted one, which is

Table 3: Performance of various algorithms on the multi-view image data in terms of ACC, NMI, F-Score, and ARI (mean ± sd), where the best performance are shown in bold font, and - represents no output.

Data	Methods	ACC(%)	NMI(%)	F-Score(%)	ARI(%)
ORL	Co-Reg	73.80±3.60	88.04±1.30	64.78±5.26	63.90±5.39
	MultiNMF	72.00±3.40	88.70±1.09	65.38±3.94	64.46±4.08
	DMVC	76.09±1.36	87.44±0.47	67.55±1.38	66.76±1.42
	CSMSC	77.63±2.65	90.40±0.98	71.99±2.52	71.28±2.59
	GMC	63.25±0.00	85.71±0.00	35.99±0.00	33.67±0.00
	MVC-DMF-PA	71.27±3.66	85.81±1.80	62.85±4.30	63.76±4.18
	CGDD	62.20±1.33	84.66±0.80	35.11±2.91	32.76±3.07
	MVC-DMLR	**81.81±2.57**	**92.49±0.84**	**76.91±2.49**	**76.34±2.55**
Yale	Co-Reg	58.09±4.20	63.06±2.45	43.78±2.26	39.93±2.43
	MultiNMF	56.95±2.89	63.70±1.50	45.06±2.26	41.20±2.50
	DMVC	**78.20±1.00**	74.50±1.10	60.10±0.20	57.90±0.20
	CSMSC	64.46±2.80	68.85±1.62	51.72±2.13	48.39±2.31
	GMC	65.45±0.00	68.92±0.00	48.01±0.00	44.10±0.00
	MVC-DMF-PA	60.08±5.90	64.52±4.66	43.59±6.44	47.24±5.97
	CGDD	66.55±0.27	70.35±1.06	49.53±1.48	45.88±1.60
	MVC-DMLR	73.16±1.09	**75.11±0.88**	**60.58±1.18**	**57.96±1.25**
100leaves	Co-Reg	71.89±1.77	88.16±0.75	61.88±8.06	61.49±8.14
	MultiNMF	66.95±1.86	85.69±0.56	58.52±1.83	58.08±1.86
	DMVC	25.99±0.44	56.56±0.25	11.87±0.29	10.96±0.30
	CSMSC	75.74±1.43	89.11±0.48	68.41±1.30	68.09±1.31
	GMC	82.38±0.00	92.92±0.00	50.42±0.00	49.74±0.00
	MVC-DMF-PA	-	-	-	-
	CGDD	80.34±0.96	92.08±0.75	46.63±10.37	45.88±10.57
	MVC-DMLR	**83.38±1.68**	**92.98±0.52**	**77.87±1.66**	**77.65±1.68**

defined as

$$\mathrm{NMI}(C, C^*) = \frac{-2\sum_{i=1}^{|C|}\sum_{j=1}^{|C^*|} p_{ij}\log\left(\frac{p_{ij}P}{P_{i.}P_{.j}}\right)}{\sum_{i=1}^{|C|} P_{i.}\log\left(\frac{P_{i.}}{P}\right) + \sum_{j=1}^{|C^*|} P_{.j}\log\left(\frac{P_{j.j}}{P}\right)}.$$

ARI [53] is defined as

$$ARI = \frac{\sum_{ij}\binom{p_{ij}}{2} - \left[\sum_i \binom{p_{i.}}{2}\sum_j \binom{p_{.j}}{2}\right]/\binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{p_{i.}}{2} + \sum_j \binom{p_{.j}}{2}\right] - \left[\sum_i \binom{p_{i.}}{2}\sum_j \binom{p_{.j}}{2}\right]/\binom{n}{2}}, \tag{33}$$

where $p_{i.}$ and $p_{.j}$ is the sum of the i-th row and j-column, respectively. F-score is variant of mean of precision and recall as

$$\text{F-score} = \frac{2 \times precision \times recall}{precision + recall}. \tag{34}$$

5.3. Parameter analysis

Two parameters λ and γ are involved in MVC-DMLR, where parameter λ and γ determine importance of diversity and consistency of features, respectively. Parameter effect is analyzed by fixing the others on six datasets.

How ACC of MVC-DMLR alters by varying values of parameters is displayed in Fig. 2, where panel A is for parameter λ, and B for γ, respectively. Fig. 2 A depicts how ACC of MVC-DMLR changes as parameter λ ranges from 10^{-4} to 10^{2} on various datasets. We can draw conclusions naturally that ACC significantly improves as λ increases from 10^{-4} to 1. Furthermore, ACC of MVC-DMLR is quite steady when $\lambda > 1$. When λ is small, diversity is subtle in the objective function, failing to balance diversity and consistency, which enforces MVC-DMLR to maximize consistency of various views. In this case, MVC-DMLR fails to separate diversity from consistency of each view, thereby reducing quality of features of objects. As parameter λ increases from 1 to 10^{2}, ACC of MVC-DMLR significantly improves because diversity and consistency reach a good balance.

Moreover, Fig. 2 B tracks how ACC of MVC-DMLR alters as parameter γ ranges from 10^{-4} to 10^{2} on various datasets. Performance of MVC-DMLR improves as parameter λ increases from 10^{-4} to 10^{-2}, and keeps stable as $\lambda \in [10^{-2}, 1]$. However, performance of MVC-DMLR decreases ACC if $\gamma >1$. When parameter γ is small, contribution of relations between diversity and consistency of various views is subtle, and MVC-DMLR is devoted to learn features of objects for each views, deviating from the conserved features of various views. When γ is large, objective function is dominated by relations between diversity and consistency, reducing importance of multi-level representation of objects. MVC-DMLR achieves the best performance when $\gamma \in [0.01, 1]$.

Finally, it is natural to ask how many levels is a good choice, i.e., how to select value for parameter τ for multi-level representation. How ACC of MVC-DMLR changes by increases parameter τ from 1 to 5 is shown in Fig. 3. ACC of MVC-DMLR increases as parameter τ increases from 1 to 3, while it decreases as τ keeps increasing. When parameter τ is small, multi-level representation cannot

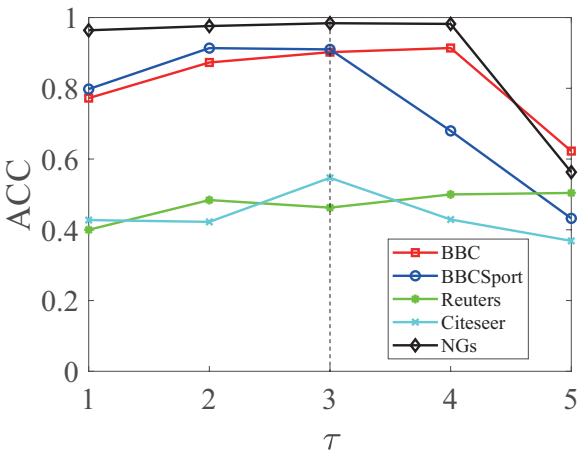

Figure 3: Performance of MVC-DMLR with various values of parameter τ.

fully characterize and model intrinsic structure of multi-view data. Furthermore, large value of parameter τ results in over-fitting, thereby reducing performance of MVC-DMLR. And, the proposed algorithm reaches a good tradeoff as τ=3. By replacing ACC with NMI and ARI, we have the similar tendency for each parameter, which are missed to remove redundancy. Therefore, we set λ=1, γ=0.1, τ=3 for all experiments.

5.4. Convergence analysis

MVC-DMLR adopts ADMM to optimize objective function, where convergence is ensured [45]. Here, by using the relative error of objective function, we investigate convergence of the suggested approach, i.e., $(\mathcal{O}_i\text{-}\mathcal{O}_{min})/(\mathcal{O}_{max}\text{-}\mathcal{O}_{min})$, where $\mathcal{O}_{min}$, $\mathcal{O}_{max}$, and $\mathcal{O}_{min}$ denotes the i-th iteration, maximal and minimal value of objective function, respectively.

Fig. 4 depicts convergence of MVC-DMLR relative error of objective function and the number of iterations for all datasets, where panel A is for BBC, B for Yale, C for ORL, and D for 3Source, respectively. Notice that the tendency is similar in other datasets, which is absent for removing redundancy. From these panels, it is easy to assert that MVC-DMLR only takes about 50 iterations to converge, whereas Multi-NMF requires more than 300 iterations to converge,

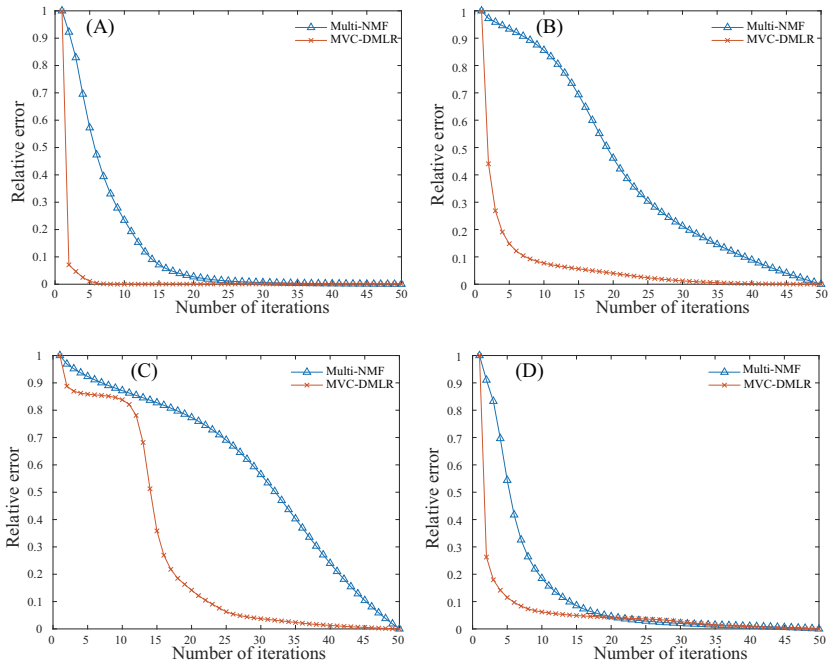

Figure 4: Convergence analysis of MVC-DMLR on various datasets: (A) BBC, (B) Yale, (C) ORL, and (D) 3Sources, respectively.

implying that the suggested approach is efficient. Three reasons explain why the proposed algorithm converges quickly. First, MVC-DMLR takes DNMF to learn multi-level representation of objects, i.e., in essence dimension reduction is performed for each level, which dramatically reduces complexity of the proposed algorithm. Second, MVC-DMLR joins feature learning, graph learning and clustering of multi-view data, where the topological structure serves as partial information to accelerate speed of convergence [40, 41, 42]. Third, MVC-DMLR simultaneously measures relations consistency and diversity of various views with orthogonality, enforcing sparsity of diversity of various views, which also accelerates speed of the proposed algorithm.

5.5. Performance of multi-view clustering

To validate performance of MVC-DMLR, seven state-of-the-art algorithms, such as Co-Reg [10], MultiNMF [51], DMVC [35], CSMSC [22], GMC [19],

MVC-DMF-PA [36], and CGDD [20], are chosen as baselines. Furthermore, ten benchmark multi-view data are selected for experiments, which are divided into two classes, i.e., image and text, as shown in Table 2. To remove bias of measurements, four indexes, including ACC, ARI, NMI and F-score, are employed to quantify performance of various algorithms. To remove randomness of algorithms, each methods are executed 50 time and mean $\pm$ standard deviation is chosen as performance.

Performance of various algorithms for clustering of multi-view image data, such as ORL, Yale and 100leaves, is shown in Table 3, where the proposed algorithm achieves the best performance. Specifically, ACC of MVC-DMLR is 81.81% for ORL, 73.16% for Yale, and 83.38% for 100leaves, whereas that of CGDD is 62.20%, 66.55%, and 80.34%, respectively. CSMSC and GMC are inferior to MVC-DMLR, but are superior to others. These results demonstrate that topological structure of graphs facilitates the identification of clusters. Furthermore, ACC of DMVC is 76.09% for ORL, whereas it is 25.99% for 100leaves, indicating that DMVC is very sensitive to datasets. The reason is that DMVC solely learns the multi-level representation of objects, which is insufficient to characterize the intrinsic structure of multi-view data. By replacing ACC with ARI, NMI and F-score, MVC-DMLR still achieves the best performance, demonstrating that the proposed algorithm is not co-factored by measurements. However, CGDD and GMC are very sensitive to measurements. For example, ACC of GMC and CGDD is 63.25% and 62.20% for ORL, whereas ARI is 33.67% and 32.76%, respectively. This tendency repeats in other datasets, showing these algorithms fail to identify the truth clusters in multi-view data. These results demonstrate that MVC-DMLR is promising for clustering of multi-view datasets for images.

Then, we apply these algorithms to multi-view data for documents, and performance of them with various measurements is shown in Table 5. Notice that even though MVC-DMLR is inferior to CGDD and GMC in terms ACC and NMI on WebKB, but it achieves the best performance for other datasets. Performance of MVC-DMLR is still acceptable for two reasons. First, difference

between MVC-DMLR and CGDD is subtle, i.e., 74.88% vs 76.22%. Second, MVC-DMLR is also superior to CGDD and GMC in terms of F-score and ARI. In all, the proposed algorithm significantly outperforms these baselines. In details, ACC of MVC-DMLR is 91.82% for BBC, 92.29% for BBCSport, 79.88% for 3sources, 57.25% for Reuters, 59.69% for CiteSeer, and 98.40% for NGs, whereas that of MVC-DMF-PA is 75.48 % for BBC, 68.54% for BBCSport, 54.89% for 3sources, 49.85% for Reuters, 26.87% for CiteSeer, and 45.66% for NGs, respectively. Furthermore, the proposed algorithm is superior to baselines in terms of F-score and ARI for all datasets, implying that MVC-DMLR avoids identifying clusters with large sizes. The reason why MultiNMF and GMC are inferior to others is that they only explore consistency of different views, failing to discover relations of various views. In contrast, MVC-DMLR, CSMSC and CGDD takes both consistency and diversity into consideration, which improves performance of clustering. Although CSMSC and CGDD learn diversity and consistency of various views, they fail to exploit multi-level representation of multi-view data, resulting in undesirable performance.

Several reasons accounts for superiority of the proposed algorithm. First, MVC-DMLR learns multi-level representation of objects with matrix factorization, where the hierarchical representation is exploited, providing a comprehensive and precise way to characterize the intrinsic structure of multi-view data. Second, MVC-DMLR learns multi-layer graphs of each view, where the indirected topological information is explored, thereby replenishing the original features of objects. Third, MVC-DMLR separates diversity and consistency of various views, thereby improving performance of multi-view clustering. These results demonstrate that the proposed multi-level topology representation is promising for multi-view clustering.

5.6. Ablation Study

Since MVC-DMLR simultaneously learns multi-level topology representation, and diversity and consistency learning, which are integrated with regularization. Thus, it is necessary to conduct an ablation study, where importance

Table 4: Performance of variants of MVC-DMRL on various datasets in terms of ACC.

Data	λ=0	γ=0	MVC-DMLR
BBC	83.18±0.08	75.13±0.07	**91.82±0.00**
BBCSport	79.60±0.00	77.38±0.05	**92.31±0.07**
ORL	55.53±1.60	75.83±1.69	**82.14±2.75**
Yale	64.07±2.18	66.51±1.11	**72.61±0.75**
Reuters	46.97±0.23	27.61±0.26	**57.25±0.00**
webkb	50.25±0.00	66.02±0.75	**75.37±0.00**
NGs	76.25±0.13	93.22±0.06	**98.40±0.00**

of these items are investigated.

Two variants of MVC-DMLR are generated by setting either λ=0 or γ=0, where the first variant removes diversity of each view, and the second one deletes relations of diversity and consistency. Performance of MVC-DMLR and its variants on all datasets in terms of ACC is shown in Table 4, where ACC of MVC-DMLR dramatically decreases by removing either of them. For example, ACC of MVC-DMLR is 91.82% for BBC, 92.31% for BBCSport, 82.14% for ORL, 72.61% for Yale, 57.25% for Reuters, 75.37% for WebKB, and 98.40% for NGs, respectively. However, it significantly drops to 83.18% for BBC, 79.16% for BBCSport, 55.53% for ORL, 64.07% for Yale, 46.97% for Reuters, 50.25% for WebKB, and 76.25% for NGs if λ=0. These results demonstrate diversity is critical for the characterization of structure of clusters in multi-view data.

Furthermore, ACC of MVC-DMLR incredibly descends to 75.13% for BBC, 77.38% for BBCSport, 75.83% for ORL, 66.51% for Yale, 27.61% for Reuters, 66.02% for WebKB, and 93.22% for NGs if relations between diversity and consistency of various views is deleted, i.e., γ=0. These results prove that relations between diversity and consistency is also critical for MVC-DMLR, indicating that diversity and consistency are complementary information for the characterization of multi-view data. And, MVC-DMLR seamlessly integrates these two issues with regularization.

6. Conclusion

Complex systems are more precisely characterized with multiple views, and the resulted in multi-view data provide an opportunity to exploit structure and

functions of systems. However, state-of-the-art methods are criticized for failing to capture intrinsic structure of features. In this study, we propose a novel multi-view clustering method, which learns the multi-level topology representation, and exploits relations among views by exploring structure consistency and diversity. Experimental results demonstrate that the proposed algorithm outperforms state-of-the-art baselines, indicating that multi-level topology structure is promising for characterizing multi-view data.

There are some possible directions for further study, which are listed as

- MVC-DMLR implicitly assumes that all views of data and levels of representation are equal. Actually, this hypothesis deviates from the expectation. How to automatically learn weights for each view and each representation level is critical for further improve performance of clustering.
- Even though the proposed method learns multi-level topology representation of multi-view data, MVC-DMLR addresses consistency and diversity of views at the graph level, ignoring roles of vertices. Actually, roles of vertices is also promising for clustering. Thus, how to exploit additional roles of vertices for multi-level topology representation of multi-view data is also interesting.
- MVC-DMLR employs NMF for feature learning of multi-view data, which is time consuming. How to accelerate the proposed algorithm for large-scale datasets is also promising. Furthermore, relations between diversity and consistency are critical to model structure of multi-view data.
- MVC-DMLR adopts self-representation learning to construct graphs for various views, which only characterizes linear relations among objects. How to model and measure non-linear relations among objects is very interesting.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to have an impact on the

research presented in this study.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 62272361) and Shaanxi Key Research and Development Program (Program No. 2021ZDLGY02-02).

[1] J. MacQueen, Some methods for classification and analysis of multivariate observations, in: Proc. 5th Berkeley Symposium on Math., Stat., and Prob, 1965, p. 281.

[2] J. A. Hartigan, M. A. Wong, Algorithm as 136: A k-means clustering algorithm, Journal of the royal statistical society. series c (applied statistics) 28 (1) (1979) 100–108.

[3] D. D. Lee, H. S. Seung, Learning the parts of objects by non-negative matrix factorization, Nature 401 (6755) (1999) 788–791.

[4] A. Ng, M. Jordan, Y. Weiss, On spectral clustering: Analysis and an algorithm, Advances in neural information processing systems 14.

[5] U. Von Luxburg, A tutorial on spectral clustering, Statistics and computing 17 (2007) 395–416.

[6] Q. Wang, Z. Qin, F. Nie, X. Li, Spectral embedded adaptive neighbors clustering, IEEE transactions on neural networks and learning systems 30 (4) (2018) 1265–1271.

[7] A. Fahad, N. Alshatri, Z. Tari, A. Alamri, I. Khalil, A. Y. Zomaya, S. Foufou, A. Bouras, A survey of clustering algorithms for big data: Taxonomy and empirical analysis, IEEE transactions on emerging topics in computing 2 (3) (2014) 267–279.

[8] C. Xu, D. Tao, C. Xu, A survey on multi-view learning, arXiv preprint arXiv:1304.5634.

[9] A. Kumar, H. Daumé, A co-training approach for multi-view spectral clustering, in: Proceedings of the 28th international conference on machine learning (ICML-11), 2011, pp. 393–400.

[10] A. Kumar, P. Rai, H. Daume, Co-regularized multi-view spectral clustering, Advances in neural information processing systems 24.

[11] S. Huang, Z. Xu, I. W. Tsang, Z. Kang, Auto-weighted multi-view co-clustering with bipartite graphs, Information Sciences 512 (2020) 18–30.

[12] G. Chen, S. Atev, G. Lerman, Kernel spectral curvature clustering (kscc), in: 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops, IEEE, 2009, pp. 765–772.

[13] G. Tzortzis, A. Likas, Kernel-based weighted multi-view clustering, in: 2012 IEEE 12th international conference on data mining, IEEE, 2012, pp. 675–684.

[14] D. Guo, J. Zhang, X. Liu, Y. Cui, C. Zhao, Multiple kernel learning based multi-view spectral clustering, in: 2014 22nd International conference on pattern recognition, IEEE, 2014, pp. 3774–3779.

[15] Z. Ren, Q. Sun, D. Wei, Multiple kernel clustering with kernel k-means coupled graph tensor learning, in: Proceedings of the AAAI conference on artificial intelligence, Vol. 35, 2021, pp. 9411–9418.

[16] J. Liu, X. Liu, Y. Yang, Q. Liao, Y. Xia, Contrastive multi-view kernel learning, IEEE Transactions on Pattern Analysis and Machine Intelligence.

[17] C. Hou, F. Nie, H. Tao, D. Yi, Multi-view unsupervised feature selection with adaptive similarity and view weight, IEEE Transactions on Knowledge and Data Engineering 29 (9) (2017) 1998–2011.

[18] X. Gao, X. Ma, W. Zhang, J. Huang, H. Li, Y. Li, J. Cui, Multi-view clustering with self-representation and structural constraint, IEEE Transactions on Big Data 8 (4) (2021) 882–893.

[19] H. Wang, Y. Yang, B. Liu, Gmc: Graph-based multi-view clustering, IEEE Transactions on Knowledge and Data Engineering 32 (6) (2019) 1116–1129.

[20] S. Huang, I. W. Tsang, Z. Xu, J. Lv, Cgdd: Multiview graph clustering via cross-graph diversity detection, IEEE Transactions on Neural Networks and Learning Systems.

[21] W. Guo, H. Che, M.-F. Leung, Tensor-based adaptive consensus graph learning for multi-view clustering, IEEE Transactions on Consumer Electronics.

[22] S. Luo, C. Zhang, W. Zhang, X. Cao, Consistent and specific multi-view subspace clustering, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 32, 2018.

[23] P. Zhang, X. Liu, J. Xiong, S. Zhou, W. Zhao, E. Zhu, Z. Cai, Consensus one-step multi-view subspace clustering, IEEE Transactions on Knowledge and Data Engineering.

[24] D. Xie, Q. Gao, M. Yang, Enhanced tensor low-rank representation learning for multi-view clustering, Neural Networks 161 (2023) 93–104.

[25] A. Dong, Z. Wu, H. Zhang, Multi-view subspace clustering based on adaptive search, Knowledge-Based Systems (2024) 111553.

[26] S.-Y. Li, Y. Jiang, Z.-H. Zhou, Partial multi-view clustering, in: Proceedings of the AAAI conference on artificial intelligence, Vol. 28, 2014.

[27] Q. Yin, S. Wu, R. He, L. Wang, Multi-view clustering via pairwise sparse subspace representation, Neurocomputing 156 (2015) 12–21.

[28] X. Cao, C. Zhang, H. Fu, S. Liu, H. Zhang, Diversity-induced multi-view subspace clustering, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 586–594.

[29] P. Ji, T. Zhang, H. Li, M. Salzmann, I. Reid, Deep subspace clustering networks, Advances in neural information processing systems 30.

[30] P. Zhu, B. Hui, C. Zhang, D. Du, L. Wen, Q. Hu, Multi-view deep subspace clustering networks, arXiv preprint arXiv:1908.01978.

[31] H. Huang, G. Zhou, Q. Zhao, L. He, S. Xie, Comprehensive multiview representation learning via deep autoencoder-like nonnegative matrix factorization, IEEE Transactions on Neural Networks and Learning Systems.

[32] Y. Guo, Convex subspace representation learning from multi-view data, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 27, 2013, pp. 387–393.

[33] H. Gao, F. Nie, X. Li, H. Huang, Multi-view subspace clustering, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 4238–4246.

[34] G. Trigeorgis, K. Bousmalis, S. Zafeiriou, B. W. Schuller, A deep matrix factorization method for learning attribute representations, IEEE transactions on pattern analysis and machine intelligence 39 (3) (2016) 417–429.

[35] H. Zhao, Z. Ding, Y. Fu, Multi-view clustering via deep matrix factorization, in: Thirty-first AAAI conference on artificial intelligence, 2017.

[36] C. Zhang, S. Wang, J. Liu, S. Zhou, P. Zhang, X. Liu, E. Zhu, C. Zhang, Multi-view clustering via deep matrix factorization and partition alignment, in: Proceedings of the 29th ACM International Conference on Multimedia, 2021, pp. 4156–4164.

[37] R. Li, C. Zhang, Q. Hu, P. Zhu, Z. Wang, Flexible multi-view representation learning for subspace clustering., in: IJCAI, 2019, pp. 2916–2922.

[38] Y. Zhao, H. Wang, J. Pei, Deep non-negative matrix factorization architecture based on underlying basis images learning, IEEE Transactions on Pattern Analysis and Machine Intelligence 43 (6) (2019) 1897–1913.

[39] E. Elhamifar, R. Vidal, Sparse subspace clustering: Algorithm, theory, and applications, IEEE transactions on pattern analysis and machine intelligence 35 (11) (2013) 2765–2781.

[40] X. Ma, D. Dong, Q. Wang, Community detection in multi-layer networks using joint nonnegative matrix factorization, IEEE Transactions on Knowledge and Data Engineering 31 (2) (2018) 273–286.

[41] Z. Huang, Y. Wang, X. Ma, Clustering of cancer attributed networks by dynamically and jointly factorizing multi-layer graphs, IEEE/ACM Transactions on Computational Biology and Bioinformatics 19 (5) (2021) 2737–2748.

[42] B. Zhang, X. Ma, Multi-view clustering with constructed bipartite graph in embedding space, Knowledge-Based Systems 254 (2022) 109690.

[43] V.-S. Chellaboina, W. M. Haddad, Is the frobenius matrix norm induced?, IEEE Transactions on Automatic Control 40 (12) (1995) 2137–2139.

[44] J. Wang, F. Tian, H. Yu, C. H. Liu, K. Zhan, X. Wang, Diverse non-negative matrix factorization for multiview data representation, IEEE transactions on cybernetics 48 (9) (2017) 2620–2632.

[45] Z. Lin, R. Liu, Z. Su, Linearized alternating direction method with adaptive penalty for low-rank representation, Advances in neural information processing systems 24.

[46] D. Lee, H. S. Seung, Algorithms for non-negative matrix factorization, Advances in neural information processing systems 13.

[47] X. Li, M. Chen, Q. Wang, Discrimination-aware projected matrix factorization, IEEE Transactions on Knowledge and Data Engineering 32 (4) (2019) 809–814.

[48] D. Greene, P. Cunningham, Practical solutions to the problem of diagonal dominance in kernel document clustering, in: Proceedings of the 23rd international conference on Machine learning, 2006, pp. 377–384.

[49] G. Bisson, C. Grimal, An architecture to efficiently learn co-similarities from multi-view datasets, in: International Conference on Neural Information Processing, Springer, 2012, pp. 184–193.

[50] D. Cai, X. He, Y. Hu, J. Han, T. Huang, Learning a spatially smooth subspace for face recognition, in: 2007 ieee conference on computer vision and pattern recognition, IEEE, 2007, pp. 1–7.

[51] J. Liu, C. Wang, J. Gao, J. Han, Multi-view clustering via joint nonnegative matrix factorization, in: Proceedings of the 2013 SIAM international conference on data mining, SIAM, 2013, pp. 252–260.

[52] L. Danon, A. Diaz-Guilera, J. Duch, A. Arenas, Comparing community structure identification, Journal of statistical mechanics: Theory and experiment 2005 (09) (2005) P09008.

[53] L. Hubert, P. Arabie, Comparing partitions, Journal of classification 2 (1985) 193–218.

Table 5: Performance of various algorithms in terms of various measurements, where the best performance are shown in bold font (mean ± sd).

Data	Methods	ACC(%)	NMI(%)	F-Score(%)	ARI(%)
BBC	Co-Reg	42.98±4.70	15.86±7.36	40.63±2.13	8.41±4.45
	MultiNMF	45.57±0.35	24.86±0.15	37.22±0.24	11.43±0.21
	DMVC	37.50±0.25	11.73±0.10	28.64±0.03	7.70±0.07
	CSMSC	91.81±0.04	**77.29±0.03**	85.52±0.08	81.05±0.11
	GMC	69.34±0.00	56.28±0.00	63.33±0.00	47.89±0.00
	MVC-DMF-PA	75.84±7.85	59.98±2.06	58.78±6.20	67.94±4.92
	CGDD	88.06±0.07	75.47±0.14	82.67±0.16	76.81±0.22
	MVC-DMLR	**91.82±0.00**	77.18±0.00	**86.00±0.00**	**81.74±0.00**
BBCSport	Co-Reg	36.27±2.75	13.93±0.45	32.98±0.59	11.66±0.81
	MultiNMF	46.63±0.86	33.10±1.83	40.18±1.45	19.14±2.84
	DMVC	33.49±0.26	6.12±0.02	26.77±0.04	3.73±0.01
	CSMSC	84.93±0.00	73.57±0.00	79.10±0.00	72.87±0.00
	GMC	80.70±0.00	76.00±0.00	79.43±0.00	72.18±0.00
	MVC-DMF-PA	68.54±4.49	49.48±3.45	46.25±4.07	58.06±3.19
	CGDD	79.96±0.64	73.12±1.79	74.00±1.40	64.01±2.01
	MVC-DMLR	**92.29±0.04**	**80.66±0.16**	**86.58±0.07**	**82.57±0.09**
3sources	Co-Reg	57.51±3.53	50.65±3.30	47.17±3.42	32.16±4.99
	MultiNMF	50.22±2.12	45.68±1.01	45.66±2.55	30.37±3.22
	DMVC	41.51±0.30	24.69±0.14	32.28±0.17	14.90±0.20
	CSMSC	63.43±1.02	47.28±1.77	63.02±0.69	50.69±0.95
	GMC	69.23±0.00	62.16±0.00	60.47±0.00	44.31±0.00
	MVC-DMF-PA	54.89±4.46	56.86±3.68	41.73±4.76	53.49±3.65
	CGDD	76.21±0.26	68.74±0.95	67.68±0.63	55.31±0.90
	MVC-DMLR	**79.88±0.00**	**74.17±0.00**	**76.51±0.00**	**69.37±0.00**
Reuters	Co-Reg	45.78±1.03	27.17±0.79	34.61±0.10	20.12±0.27
	MultiNMF	20.74±1.85	9.37±1.12	28.34±0.07	0.55±0.68
	DMVC	27.57±0.31	12.76±0.16	28.72±0.06	5.95±0.09
	CSMSC	44.51±0.06	23.59±0.04	32.41±0.03	16.84±0.05
	GMC	19.92±0.00	13.51±0.00	28.76±0.00	1.33±0.00
	MVC-DMF-PA	49.85±1.90	30.28±1.45	23.40±1.56	**37.71±0.95**
	CGDD	23.70±0.05	20.22±0.36	29.29±0.07	3.03±0.13
	MVC-DMLR	**57.25±0.00**	**38.14±0.00**	**43.39±0.00**	30.85±0.00
CiteSeer	Co-Reg	40.42±1.01	18.76±0.35	32.64±2.21	15.05±7.84
	MultiNMF	22.34±0.00	2.90±0.03	30.20±0.00	0.12±0.00
	DMVC	20.92±0.00	2.20±0.00	30.03±0.00	-0.14±0.00
	CSMSC	59.45±0.03	31.33±0.03	43.13±0.03	31.06±0.04
	GMC	21.44±0.00	1.41±0.00	30.31±0.00	0.11±0.00
	MVC-DMF-PA	26.87±1.59	7.33±1.38	0.99±1.55	29.28±0.43
	CGDD	21.04±0.00	2.34±0.00	30.16±0.00	-0.03±0.00
	MVC-DMLR	**59.69±0.01**	**32.15±0.01**	**43.56±0.00**	**31.31±0.01**
WebKB	Co-Reg	63.30±1.63	33.31±2.41	57.80±1.72	36.03±2.26
	MultiNMF	74.88±0.00	34.66±0.00	68.38±0.00	44.65±0.00
	DMVC	46.31±0.00	12.28±0.00	42.99±0.00	11.34±0.00
	CSMSC	67.98±0.00	32.55±0.00	61.38±0.00	36.91±0.00
	GMC	75.86±0.00	**42.19±0.00**	68.57±0.00	41.13±0.00
	MVC-DMF-PA	55.04±0.88	11.37±1.83	15.54±3.35	49.05±0.77
	CGDD	**76.26±0.41**	38.06±1.85	69.00±0.49	45.44±0.73
	MVC-DMLR	74.88±0.00	40.47±0.00	**70.88±0.00**	**49.23±0.00**
NGs	Co-Reg	24.83±2.89	4.63±2.27	32.68±0.36	0.30±0.49
	MultiNMF	30.80±0.00	15.09±0.00	32.99±0.00	2.99±0.00
	DMVC	39.56±0.11	15.52±0.09	34.22±0.07	12.65±0.05
	CSMSC	98.20±0.00	93.92±0.00	96.43±0.00	95.54±0.00
	GMC	98.20±0.00	93.92±0.00	96.43±0.00	95.54±0.00
	MVC-DMF-PA	45.66±2.43	23.94±3.15	14.90±1.67	34.72±1.60
	CGDD	97.68±0.11	92.41±0.25	95.38±0.21	94.23±0.27
	MVC-DMLR	**98.40±0.00**	**94.61±0.00**	**96.81±0.00**	**96.02±0.00**