Clustering-Induced Adaptive Structure Enhancing Network for Incomplete Multi-View Data

Zhe Xue[1] , **Junping Du**[1*] , **Changwei Zheng**[1] , **Jie Song**[1] , **Wenqi Ren**[2] and **Meiyu Liang**[1]
[1] Beijing Key Laboratory of Intelligent Telecommunication Software and Multimedia, School of Computer Science, Beijing University of Posts and Telecommunications, Beijing 100876, China
[2] State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China
{xuezhe, zhengchangwei, songs, meiyu1210}@bupt.edu.cn, junpingdu@126.com, renwenqi@iie.ac.cn

Abstract

Incomplete multi-view clustering aims to cluster samples with missing views, which has drawn more and more research interest. Although several methods have been developed for incomplete multi-view clustering, they fail to extract and exploit the comprehensive global and local structure of multi-view data, so their clustering performance is limited. This paper proposes a Clustering-induced Adaptive Structure Enhancing Network (CASEN) for incomplete multi-view clustering, which is an end-to-end trainable framework that jointly conducts multi-view structure enhancing and data clustering. Our method adopts multi-view autoencoder to infer the missing features of the incomplete samples. Then, we perform adaptive graph learning and graph convolution on the reconstructed complete multi-view data to effectively extract data structure. Moreover, we use multiple kernel clustering to integrate the global and local structure for clustering, and the clustering results in turn are used to enhance the data structure. Extensive experiments on several benchmark datasets demonstrate that our method can comprehensively obtain the structure of incomplete multi-view data and achieve superior performance compared to the other methods.

1 Introduction

Describing objects from different aspects constitutes multi-view data. For instance, an image can be described by different features such as color, texture, surrounding texts or deep features. The content of a web page can be described by text, images, and videos, etc. Each view contains some specific information that other views do not have. Therefore, leveraging multiple views can obtain multi-view complementary information and generate more complete descriptions of data. Multi-view clustering aims to exploit diverse and complementary features of different views to improve clustering performance, and many methods have been developed in the past few years [Kang *et al.*, 2020; Zhang *et al.*, 2020b; Li *et al.*, 2020; Kang *et al.*, 2021].

In real-world applications, it is often the case that each view suffers from loss of information due to data collection equipment failures or environmental changes [Li *et al.*, 2014; Zhang *et al.*, 2020a]. For instance, different types of tests may be conducted for different medical applications, which leads to the loss of some views. However, conventional multi-view clustering methods assume that all views are available for each sample so that they cannot well handle the incomplete multi-view data. It is more challenging and meaningful to endow the method with adaptive capability and high effectiveness for view-missing situations.

To discover clusters from incomplete multi-view data, a variety of *incomplete multi-view clustering* (IMC) methods have been developed [Peng *et al.*, 2019; Yang *et al.*, 2021; Huang *et al.*, 2020]. Some methods adopt matrix factorization model to extract a consensus matrix from incomplete multi-view data. PVC [Li *et al.*, 2014] establishes a latent subspace where the instances corresponding to the same sample in different views are close to each other. Several methods further extend PVC method. IMG [Zhao *et al.*, 2016] adopts a graph Laplacian term to couple the incomplete multi-view samples. MIC [Shao *et al.*, 2015] integrates weighted nonnegative matrix factorization and $l_{2,1}$ regularization to handle the incomplete multi-view data. Moreover, some methods adopt multiple graph learning or multiple kernel learning for IMC [Wen *et al.*, 2019; Zhou *et al.*, 2019]. OPIMC [Hu and Chen, 2019b] directly obtains clustering results for large-scale multi-view data through a regularized matrix factorization model. PIC [Wang *et al.*, 2019] adopts spectral perturbation theory and learns a consensus Laplacian matrix from incomplete multi-view data for clustering. MKKMIK [Liu *et al.*, 2019] integrates kernel imputation and clustering into a unified learning procedure, where incomplete kernels can be adaptively imputed and combined for clustering.

In addition to the shallow models mentioned above, some deep learning based methods are proposed for IMC [Wang *et al.*, 2018; Wen *et al.*, 2020b]. AIMC [Xu *et al.*, 2019] learns the consensus latent space and performs missing data inference simultaneously, where a generative adversarial network is used to infer missing data. CDIMC-net [Wen *et al.*, 2020a] incorporates view-specific deep encoders and graph embedding into a framework to capture the local structure of each

*Corresponding Author

view, and a self-paced strategy is adopted to train the deep model. Deep learning based IMC methods can well handle the discrepancy and dependence among multiple views, so they achieve better clustering performance and wider application prospects than the shallow models.

Despite significant progress has been made in the field of incomplete multi-view clustering, there are still some issues that have not been well solved. First, it is difficult to obtain the complete data structure when some views are missing. Most of the existing IMC methods fail to complete the missing features to extract the intrinsic data structure. Second, both global and local structures are essential for IMC, yet most of the existing methods fail to simultaneously use them so that the data structure information cannot be comprehensively exploited. Third, the reliability and accuracy of views are different, so they should have different importance during the clustering process. However, the existing deep learning based IMC methods ignore this factor and their clustering performance are limited.

In order to solve the above issues, we propose a novel incomplete multi-view clustering method, *i.e.*, Clustering-induced Adaptive Structure Enhancing Network (CASEN). We develop an end-to-end trainable framework for joint multi-view structure enhancing and data clustering. CASEN is composed of a multi-view autoencoder module, an adaptive multi-view graph structure extraction module and a clustering-induced structure enhancing module. We introduce multi-view autoencoder to extract the global structure of multi-view data as well as infer the missing features of incomplete samples. Then, we leverage adaptive graph learning and graph convolution networks (GCNs) to extract and encode the local structure of data. Our method can accurately extract local structure of data by learning suitable graph representation that best serves the clustering task. To obtain robust and reliable clustering results, we adopt multiple kernel clustering to assign different weights to views and integrate both global and local structures. The clustering results are used to supervise the network training through the self-supervision strategy so that the learned data structures can be further enhanced. We conduct comprehensive experiments on several benchmark datasets to study the properties of the proposed method. Experimental results show that CASEN consistently outperforms the state-of-the-art IMC methods, which demonstrates the advantages of our method. The main contributions of this work are summarized as follows:

- We develop a novel deep learning framework to simultaneously reconstruct the missing views and learn data structure from the reconstructed complete multi-view data. Our method can explore and extract comprehensive data structure from incomplete multi-view data to reduce the impact of missing views on structure learning.
- We exploit both global and local structures to reveal the complex relationship and intrinsic distribution of multi-view data. The global and local structure information can be effectively preserved and encoded into the latent representation of the network, which can yield better clustering performance.
- We introduce multiple kernel clustering to obtain more reliable clustering results, which is achieved by assigning different weights to views according to their importance. The multiple kernel clustering and structure learning can promote each other during network training, so the learned structures can be further enhanced.

2 Methodology

CASEN is an end-to-end multi-view clustering network which is composed of three modules as shown in Figure 1. Multi-view autoencoder module and adaptive multi-view graph structure extraction module learn the global and local structure, respectively. Clustering-induced structure enhancing module utilizes multiple kernel clustering to obtain clustering results and supervises the training of the network. We introduce the details of CASEN in the following parts.

2.1 Problem Definition

Incomplete multi-view data with V views and n samples can be denoted by a set of matrices $\{X^{(v)}\}_{v=1}^{V}$, where $X^{(v)} = [x_1^{(v)}, x_2^{(v)}, ..., x_n^{(v)}]^T \in R^{n*m_v}$, m_v is the feature dimensions of the v-th view. If the i-th sample $x_i = \{x_i^{(1)}, x_i^{(2)}, ..., x_i^{(V)}\}$ loses the features of the j-th view, then $x_i^{(j)}$ is filled by zeros. Assume that each sample would not lose the features of all the views. Our objective is to cluster n unlabeled multi-view data samples into c categories.

2.2 Multi-View Autoencoder Module

Different views have different physical meanings and they are not directly comparable. Therefore, we introduce multi-view autoencoder to obtain the common latent representation of multi-view data. By training multi-view autoencoder to encode and decode data, the latent representation can well explore the global structure of multi-view data and fully preserve multi-view complementary information. Additionally, multi-view autoencoder can be used to infer missing views of incomplete samples by the reconstruction process, which enables the model to learn the complete graph representation of multi-view data.

Multi-view autoencoder consists of several view-specific encoders $\{f^{(v)}\}_{v=1}^{V}$ and corresponding decoders $\{g^{(v)}\}_{v=1}^{V}$. To obtain the common representation of multi-view data, we make the encoders of each view $\{f^{(v)}\}_{v=1}^{V}$ share the same top hidden layer so that they have the same output. When the sample x_i is input into encoders, h_i is the output of the encoders which provides the common representation of x_i. The decoders of each view take h_i to reconstruct original data. The reconstruction process of decoders is $\widehat{x}_i^{(v)} = g^{(v)}(h_i), \forall v = 1, ..., V$. Noted that decoders can generate the features of all the views, thus the missing views of incomplete samples can be infered. The reconstructed data is denoted by $\widehat{X}^{(v)} = [\widehat{x}_1^{(v)}, \widehat{x}_2^{(v)}, ..., \widehat{x}_n^{(v)}]^T \in \mathbb{R}^{n*m_v}$. The loss function of multi-view autoencoder is defined as:

$$\mathcal{L}_R = \frac{1}{2n} \sum_{v=1}^{V} ||X^{(v)} - P^{(v)} \widehat{X}^{(v)}||_F^2 \tag{1}$$

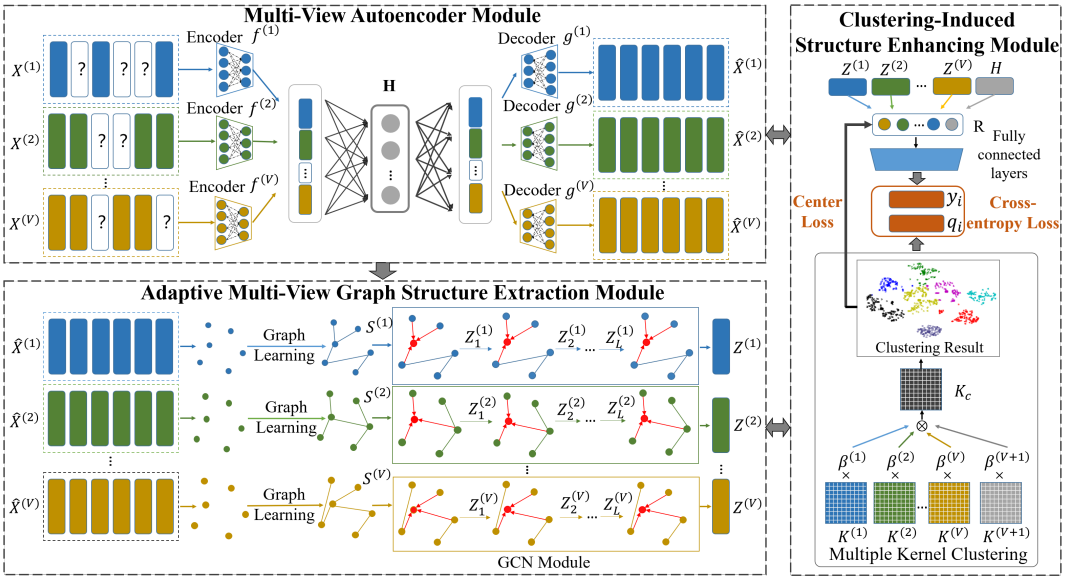


Figure 1: Architecture of the proposed Clustering-Induced Adaptive Structure Enhancing Network (CASEN).

where $P^{(v)} \in R^{n*n}$ is a diagonal matrix which records the view missing information. $P^{(v)}_{ii} = 1$ if the i-th sample is available in the v-th view, otherwise, $P^{(v)}_{ii} = 0$. By training the multi-view autoencoder to reconstruct data, the learned latent representation $H = [h_1, h_2, ..., h_n]^T \in \mathbb{R}^{n*k_h}$ can capture the global structure of multi-view data and provide effective representation for data clustering.

2.3 Adaptive Multi-View Graph Structure Extraction Module

Although the multi-view autoencoder module is able to capture the global structure of multi-view data, it ignores the local relationship of samples. In the adaptive multi-view graph structure extraction module, we introduce adaptive graph structure learning to learn graph representation of multi-view data, and leverage GCNs to further extract and explore the local structure information of multi-view data.

Given the reconstructed multi-view data $\{\widehat{X}^{(v)}\}_{v=1}^{V}$, we aim to learn a graph with affinity matrix $S^{(v)} \in \mathbb{R}^{n*n}$ for each view to represent the pairwise relationship between samples. Inspired by [Jiang *et al.*, 2019], we implement adaptive graph learning via v single-layer neural networks, which are parameterized by the weight vector $a^{(v)} \in \mathbb{R}^{m_v*1}$. Let $S^{(v)}_{ij}$ represent the similarity between x_i and x_j for the v-th view which is learned by

$$S^{(v)}_{ij} = \frac{\exp(\sigma(\mathrm{a}^{(v)^{\mathrm{T}}}|\widehat{x}^{(v)}_i - \widehat{x}^{(v)}_j|))}{\sum_{k=1}^{n}\exp(\sigma(\mathrm{a}^{(v)^{\mathrm{T}}}|\widehat{x}^{(v)}_i - \widehat{x}^{(v)}_k|))} \tag{2}$$

where σ is the activation function. Softmax operation on each row of $S^{(v)}$ can guarantee the learned graph satisfying the following property

$$\sum\nolimits_{j=1}^{n} S^{(v)}_{ij} = 1,\ S^{(v)}_{ij} \geq 0 \tag{3}$$

To ensure the graph can well capture the local structure of multi-view data, we adopt the following loss function to learn $S^{(v)}$ and the weight vectors $\{a^{(v)}\}_{v=1}^{V}$,

$$\mathcal{L}_G = \frac{1}{n}\sum_{v=1}^{V}(\sum_{i,j=1}^{n}||\widehat{x}^{(v)}_i - \widehat{x}^{(v)}_j||_2^2 S^{(v)}_{ij} + \lambda||S^{(v)}||_F^2) \tag{4}$$

where λ is the tradeoff parameter to control the sparsity of learned graph $S^{(v)}$. By adjusting the sparsity of $S^{(v)}$, we can establish the neighborhood relationship and local structure of multi-view data adaptively.

Next, we introduce GCN to further extract the local structure of multi-view data and encode them into the latent representation. For the v-th view, $Z^{(v)}_l$ is the representation learned by the l-th layer of GCN, which can be obtained by the following operation,

$$Z^{(v)}_l = \sigma({D^{(v)}}^{-1/2} S^{(v)} {D^{(v)}}^{-1/2} Z^{(v)}_{l-1} W^{(v)}_{l-1}) \tag{5}$$

where $l \in \{1, ..., L\}$, $D^{(v)}$ is a diagonal matrix with diagonal element $D^{(v)}_{ii} = \sum_{j=1}^{n} S^{(v)}_{ij}$, $W^{(v)}_{l-1}$ is the weight matrix of the convolution layer, σ is the activation function. We set the reconstructed multi-view feature as the initial node feature of GCN, *i.e.*, $Z^{(v)}_0 = \widehat{X}^{(v)}$. The last layer of GCN $Z^{(v)}_L$ is denoted by $Z^{(v)} \in \mathbb{R}^{n*k_c}$ for simplicity. GCN is able to encode both the local structure and node features. Therefore, the local structure of multi-view data can be effectively encoded into the learned representation $\{Z^{(v)}\}_{v=1}^{V}$.

2.4 Clustering-Induced Structure Enhancing Module

Now, we present the clustering-induced structure enhancing module to achieve multi-view clustering. Considering the reliability of views are different, we adopt multiple kernel clus-

tering (MKC) [Tzortzis and Likas, 2012] which assigns different weights to views so that more accurate clustering results can be obtained. Then, the clustering results are used to enhance the learned structures through a self-supervision strategy. Our method jointly implements multiple kernel clustering and structure enhancing so that they can promote each other and the clustering performance can be further improved.

In multiple kernel clustering, $\mathcal{K}(\cdot,\cdot)$ is the kernel function. The kernel matrix for each view is constructed by $K^{(v)} = \mathcal{K}(Z^{(v)}, Z^{(v)})$, and another kernel is constructed to capture the global structure by $K^{(V+1)} = \mathcal{K}(H, H)$. The unified kernel is defined by $K_u = \sum_{v=1}^{V+1} \beta^{(v)^r} \mathrm{K}^{(v)}$, where $\beta = [\beta^{(1)}, \beta^{(2)}, ..., \beta^{(V)}]$ is the weight parameter of kernels and r controls the sparsity of β. The objective of MKC is presented as follows,

$$\begin{aligned} &\min_{Q,\beta} Tr(K_u(I_n - QQ^T)) \\ &s.t. \quad Q^TQ = I_c, \beta^T \mathbf{1}_{V+1} = 1, \beta \in \mathbb{R}_+^{V+1} \end{aligned} \tag{6}$$

where $Q \in \mathbb{R}^{n*c}$ is the embedding matrix to be learned. The detailed algorithm for solving (6) is shown in Algorithm 2. By performing k-means clustering on Q, we can obtain the clustering results $q_i \in \{0,1\}^c$ for each sample x_i.

We adopt self-supervision strategy to utilize the clustering results to guide the network training. The output of multi-view autoencoder and GCNs are concatenated by $R = [H||Z^{(1)}||Z^{(2)}||...||Z^{(V)}]$, and then we put R into FC layers. The output of FC layers is denoted by $\{y_i \in \mathbb{R}^c\}_{i=1}^n$. We use self-supervision strategy to train the whole network by integrating cross-entropy loss and center loss [Wen *et al.*, 2016] as follows:

$$\mathcal{L}_C = \frac{1}{n}\sum_{i=1}^{n}(\ln(1 + e^{-\overline{y}_i^T q_i}) + \theta||r_i - \rho_{\phi_i}||_2^2) \tag{7}$$

where $\overline{y}_i$ is a normalization of y_i via softmax. r_i is the i-th row of R, which represents the concatenated representation of the i-th sample. ϕ_i takes the cluster index of q_i, and ρ_{ϕ_i} is the cluster center which corresponds to the i-th sample. θ is the tradeoff parameter. The clustering result q_i provides the pseudo labels for network training. Noted that the label index assigned to a cluster undergoes an unknown permutation during clustering. Hence, the class labels from two successive clusterings may be inconsistent. To address this issue, Hungarian algorithm [Munkres, 1957] is adopted to find an optimal assignment between the pseudo labels of successive iterations and then feed them into loss function (7).

2.5 Overall Loss Function

The overall loss function of CASEN is proposed by integrating multi-view autoencoder module, adaptive multi-view graph structure extraction module and clustering-induced structure enhancing module. By putting together the loss functions in (1), (4), and (7), the end-to-end trainable framework is formulated as follows,

$$\mathcal{L} = \mathcal{L}_R + \eta_1\mathcal{L}_G + \eta_2\mathcal{L}_C \tag{8}$$

where η_1 and η_2 are tradeoff parameters to control the importance of each component. By optimizing (8), our network is capable of jointly enhancing multi-view data structure and achieve effective clustering results.

Algorithm 1: The learning procedure of CASEN.

Input: Input data $\{X^{(v)}\}_{v=1}^V$, tradeoff parameters, network parameters, T_{max}, T_1, $iter$=1.
Output: The trained network and clustering results $\{q_i\}_{i=1}^n$.

```
Pre-train the multi-view autoencoder and initialize all the parameters of the network.
Perform MKC to get initial clustering results.
while iter ≤ T_max do
    Given the clustering results, update the network T_1 epoches by optimizing the overall loss function (8).
    Perform MKC to update clustering results by Algorithm 2.
    iter ← iter + 1.
end
```

Algorithm 2: The learning procedure of MKC.

Input: $\{K^{(v)}\}_{v=1}^{V+1}$, r, c.
Output: Clustering results $\{q_i\}_{i=1}^n$.

```
Initialize β^(v) = 1/(V+1).
while not converged do
    Update K_u = Σ_{v=1}^{V+1} β^(v)^r K^(v).
    Update Q by c largest eigenvectors of K_u.
    Update d^(v) = Tr(K^(v)(I − QQ^T)), v ∈ {1, ..., V+1}.
    Update β^(v) = 1/Σ_{v'=1}^{V+1} (d^(v)/d^(v'))^{1/(r−1)}, v ∈ {1, ..., V+1}.
end
Obtain clustering results {q_i}_{i=1}^n by performing k-means on Q.
```

2.6 Implementation

Pre-training the network. Before training the whole network, we pre-train multi-view autoencoder by using loss function (1). All features of the missing views are filled with zero. Stochastic gradient descent (SGD) is adopted to pre-train the multi-view autoencoders.

MKC model learning. For multiple kernel clustering, linear kernel is adopted as the kernel function for its simplicity and effectiveness. An iterative algorithm that alternates between updating the embedding matrix Q and updating the kernel weights β is adopted. The updating rules for Q and β can be derived by Lagrange multiplier method. The detailed learning process of MKC is shown in Algorithm 2.

Training the whole network. After pre-training the network, we use the overall loss function (8) to train the whole network. Specifically, given the clustering result, we update the other parameters in the network for T_1 epoches. Then, we perform multiple kernel clustering to update the clustering result. The two steps are performed alternately until the network is well trained. The initial clustering result is achieved by performing MKC on H, which is obtained from the pre-trained multi-view autoencoder. We present the detailed training procedure of CASEN in Algorithm 1.

3 Experiments

3.1 Experimental Settings

Datasets. We adopt four well-known multi-view learning datasets to demonstrate the effectiveness of the proposed

		ACC				NMI			
Database	Method\$p\%$	0.1	0.3	0.5	0.7	0.1	0.3	0.5	0.7
BBC	BestSV	38.52 ± 1.53	36.12 ± 1.45	31.56 ± 1.34	25.52 ± 1.82	24.99 ± 2.33	23.41 ± 2.64	18.81 ± 1.60	17.74 ± 3.47
	MIC	57.75 ± 2.30	54.25 ± 0.53	44.00 ± 3.36	35.75 ± 1.41	61.00 ± 0.53	53.50 ± 0.89	48.75 ± 0.91	37.75 ± 0.58
	OMVC	46.78 ± 3.72	38.05 ± 4.29	29.86 ± 4.03	18.18 ± 0.87	50.02 ± 2.62	42.65 ± 2.26	33.35 ± 3.51	21.31 ± 3.78
	IMG	47.04 ± 2.16	44.70 ± 1.73	41.57 ± 1.98	39.95 ± 2.24	68.23 ± 1.05	66.12 ± 1.28	62.25 ± 1.54	58.77 ± 1.70
	DAIMC	56.85 ± 3.47	48.43 ± 2.81	36.73 ± 2.88	24.80 ± 1.58	38.74 ± 2.41	31.77 ± 2.83	27.76 ± 2.55	18.50 ± 3.17
	UEAF	55.17 ± 3.41	49.67 ± 1.18	35.17 ± 2.13	22.62 ± 1.52	73.75 ± 1.41	65.67 ± 0.70	48.89 ± 6.81	33.16 ± 3.17
	OPIMC	59.60 ± 2.13	47.76 ± 2.42	36.63 ± 2.56	23.09 ± 2.17	54.95 ± 1.74	45.55 ± 2.04	37.08 ± 1.26	30.45 ± 1.02
	PIC	62.23 ± 2.45	57.13 ± 2.57	49.16 ± 1.63	43.39 ± 2.59	74.31 ± 0.88	68.94 ± 1.09	59.04 ± 0.74	49.12 ± 0.69
	CASEN	$\mathbf{63.54 \pm 1.89}$	$\mathbf{59.50 \pm 1.47}$	$\mathbf{50.25 \pm 1.51}$	$\mathbf{46.32 \pm 1.57}$	$\mathbf{75.48 \pm 0.87}$	$\mathbf{72.39 \pm 1.12}$	$\mathbf{68.18 \pm 1.37}$	$\mathbf{63.25 \pm 2.04}$
Caltech20	BestSV	43.66 ± 2.46	38.34 ± 1.65	33.54 ± 2.07	30.38 ± 0.95	51.44 ± 1.46	39.16 ± 0.61	34.99 ± 0.61	28.91 ± 0.43
	MIC	32.10 ± 1.86	26.93 ± 2.36	24.90 ± 2.50	20.33 ± 3.01	38.50 ± 3.41	33.99 ± 2.92	29.44 ± 3.17	24.68 ± 3.68
	OMVC	41.32 ± 0.81	31.39 ± 0.93	31.85 ± 1.51	22.49 ± 2.33	41.78 ± 0.98	27.02 ± 0.99	28.43 ± 2.25	14.90 ± 2.16
	IMG	46.40 ± 1.74	43.60 ± 2.43	42.63 ± 2.59	38.06 ± 3.16	58.96 ± 1.84	54.55 ± 1.62	53.04 ± 2.27	47.69 ± 2.46
	DAIMC	43.26 ± 3.61	42.82 ± 2.27	40.26 ± 2.96	34.57 ± 3.05	59.05 ± 1.83	58.28 ± 1.22	53.23 ± 0.86	38.52 ± 1.38
	UEAF	33.78 ± 1.54	32.36 ± 0.98	30.84 ± 1.86	19.28 ± 2.40	34.94 ± 1.33	32.11 ± 1.10	30.24 ± 1.59	19.23 ± 2.76
	OPIMC	55.57 ± 3.62	50.23 ± 2.30	42.21 ± 2.34	23.24 ± 3.19	47.46 ± 5.48	50.15 ± 4.12	48.20 ± 3.08	30.16 ± 6.35
	PIC	54.27 ± 2.62	53.90 ± 2.55	53.11 ± 2.37	48.77 ± 2.35	60.21 ± 3.59	61.87 ± 1.14	60.03 ± 2.11	56.92 ± 2.52
	CASEN	$\mathbf{64.66 \pm 1.20}$	$\mathbf{60.29 \pm 1.63}$	$\mathbf{57.88 \pm 1.16}$	$\mathbf{50.25 \pm 2.24}$	$\mathbf{65.45 \pm 1.05}$	$\mathbf{63.31 \pm 1.32}$	$\mathbf{61.06 \pm 1.84}$	$\mathbf{58.24 \pm 2.21}$
Wikipedia	BestSV	45.02 ± 0.23	41.34 ± 0.64	33.99 ± 0.54	24.38 ± 1.06	49.21 ± 0.66	40.26 ± 0.76	31.91 ± 0.16	24.45 ± 3.14
	MIC	48.67 ± 1.33	46.43 ± 1.04	45.93 ± 1.50	42.45 ± 2.77	37.80 ± 0.25	36.18 ± 1.39	35.40 ± 1.25	30.90 ± 1.41
	OMVC	44.54 ± 2.03	38.92 ± 1.56	32.84 ± 2.15	26.39 ± 1.94	45.21 ± 1.42	39.29 ± 1.78	33.31 ± 1.67	27.21 ± 1.55
	IMG	51.52 ± 1.43	47.77 ± 1.48	45.87 ± 1.62	42.47 ± 2.42	51.06 ± 1.21	46.29 ± 2.13	42.35 ± 1.87	40.00 ± 1.39
	DAIMC	56.04 ± 0.99	45.26 ± 1.56	33.24 ± 1.28	20.69 ± 1.70	45.95 ± 0.77	29.45 ± 1.02	18.41 ± 1.57	11.32 ± 1.46
	UEAF	54.67 ± 1.64	45.23 ± 1.70	35.11 ± 1.32	26.38 ± 1.26	50.92 ± 1.09	39.74 ± 1.62	28.32 ± 1.68	19.62 ± 1.03
	OPIMC	46.10 ± 0.95	30.11 ± 1.83	17.87 ± 0.95	10.15 ± 1.14	55.60 ± 1.17	43.18 ± 3.26	34.96 ± 1.33	27.54 ± 1.79
	PIC	45.72 ± 0.12	41.29 ± 0.63	30.93 ± 0.12	27.11 ± 0.37	34.60 ± 0.18	29.11 ± 0.27	15.14 ± 0.12	11.67 ± 0.30
	CASEN	$\mathbf{58.33 \pm 0.87}$	$\mathbf{50.54 \pm 1.10}$	$\mathbf{46.68 \pm 1.74}$	$\mathbf{42.59 \pm 1.88}$	$\mathbf{57.76 \pm 0.93}$	$\mathbf{49.13 \pm 1.14}$	$\mathbf{44.65 \pm 1.45}$	$\mathbf{41.87 \pm 1.95}$
MNIST	BestSV	44.73 ± 0.34	40.18 ± 0.29	29.06 ± 0.85	22.83 ± 2.31	38.75 ± 0.29	37.90 ± 0.26	27.34 ± 1.01	23.41 ± 2.73
	MIC	61.75 ± 2.65	54.28 ± 1.99	48.19 ± 0.35	41.38 ± 1.06	62.03 ± 0.49	55.72 ± 0.25	50.34 ± 0.16	37.75 ± 0.58
	OMVC	60.50 ± 2.49	55.79 ± 1.97	49.88 ± 1.66	37.52 ± 2.16	58.29 ± 2.52	53.16 ± 1.80	46.87 ± 1.75	35.68 ± 1.91
	IMG	55.70 ± 1.31	50.37 ± 1.69	43.62 ± 2.02	36.84 ± 2.43	67.50 ± 1.02	60.69 ± 0.85	53.56 ± 1.72	40.28 ± 1.88
	DAIMC	63.48 ± 1.30	51.53 ± 1.49	32.79 ± 0.93	20.49 ± 1.04	65.97 ± 0.59	49.29 ± 0.85	37.12 ± 0.96	25.86 ± 1.11
	UEAF	73.77 ± 2.17	54.34 ± 1.75	47.62 ± 2.53	34.05 ± 1.76	68.14 ± 1.84	47.09 ± 2.57	36.12 ± 3.90	29.29 ± 1.87
	OPIMC	61.60 ± 4.42	44.77 ± 3.17	26.15 ± 2.17	18.09 ± 2.17	65.92 ± 4.74	51.88 ± 4.91	32.59 ± 3.86	23.45 ± 1.02
	PIC	65.47 ± 0.79	60.26 ± 1.17	50.44 ± 0.77	41.52 ± 2.03	70.85 ± 0.33	66.18 ± 0.48	53.11 ± 0.38	40.57 ± 0.56
	CASEN	$\mathbf{77.13 \pm 1.29}$	$\mathbf{68.34 \pm 1.32}$	$\mathbf{56.63 \pm 1.58}$	$\mathbf{44.31 \pm 1.69}$	$\mathbf{76.36 \pm 0.87}$	$\mathbf{67.86 \pm 1.47}$	$\mathbf{55.02 \pm 1.86}$	$\mathbf{41.59 \pm 1.35}$

Table 1: Clustering performance (mean $\pm$ standard deviation) on the benchmark datasets. Bold font shows the best performance.

method. 1) **BBC**: It consists of 685 documents from BBC news website which corresponds to stories about five topical areas. Each sample is described by four views [Greene and Cunningham, 2006]. 2) **Caltech20**: It is a subset of Caltech101 dataset, which consists of 2386 images of 20 classes. To obtain multiple views, we manually extract six kinds of visual features as in [Cai *et al.*, 2013]. 3) **Wikipedia**: It contains 2866 multimedia documents which are collected from Wikipedia [Rasiwasia *et al.*, 2010]. Each document contains two views *i.e.*, the image view and the text view. 4) **MNIST**: It is composed of 10000 samples of ten digits. Pixel feature and edge feature are adopted as two views [LeCun *et al.*, 1998]. To construct the incomplete multi-view data, for data with more than two views, we randomly remove $p\%$ instances from each view while guarantee that each sample at least have one view. Wikipedia and MNIST datasets have only two views, we randomly choose $1 - p\%$ samples and keep their views complete, and the remaining samples are treated as single view samples. Half of the single view samples have the first view and the others have the second view.

Baseline methods. The compared methods include several representative IMC methods. **BestSV**: We reports the best clustering results achieved by performing k-means on each view. **MIC** [Shao *et al.*, 2015]: It learns the latent feature matrices for all the views and generates a consensus matrix so that the difference between each view is minimized. **OMVC** [Shao *et al.*, 2016]: It learns the latent feature matrices for all the views by pushing them towards a consensus. **IMG** [Zhao *et al.*, 2016]: It imposes the orthogonal constraint on the basis matrix of each view to handle the out-of-sample problem. **DAIMC** [Hu and Chen, 2019a]: It is based on weighted semi-nonnegative matrix factorization to obtain cluster results. **UEAF** [Wen *et al.*, 2019]: A locality-preserved reconstruction term is introduced to infer the missing views so that all views can be aligned. **OPIMC** [Hu and Chen, 2019b]: It adopts regularized and weighted matrix factorization to obtain clustering results. **PIC** [Wang *et al.*, 2019]: It learns a consensus Laplacian matrix from incomplete multi-view data for clustering. We adopt two widely used evaluation metrics: Clustering accuracy (ACC) and normalized mutual information (NMI) to validate the clustering performance.

Parameter settings. The autoencoders $f^{(v)}$ and $g^{(v)}$ are stacked by four layers and the dimensions are with $[0.8m_v, 0.8m_v, 1200, 50]$ and $[50, 0.8m_v, 0.8m_v, m_v]$, respectively. We adopt two convolution layers in GCN and the dimensions are $[0.8m_v, 50]$. The FC layers in $\mathcal{L}_C$ are designed with four layers $[l, d_1, d_2, d_3]$, l is the dimension of the input layer, d_1 and d_2 are the dimensions of hidden layers, d_3 is the dimension of output layer. We set $d_1 = n$, $d_2 = 0.8n$, $d_3 = c$. The other parameters are set as $\eta_1 = 0.1$, $\eta_2 = 0.01$, $\lambda = 0.5$, $\theta = 0.1$, $r = 2$. Rectified Linear Unit (ReLU) is adopted as the activation function of our network. We run all the methods 10 times and report the average per-

formance. CASEN is implemented in PyTorch and executed on an Ubuntu 18.04 machine with Nvidia GeForce RTX 2080 Ti GPU.

3.2 Performance Comparison

All the methods are conducted on four benchmark datasets with different missing rates $p \in \{10\%, 30\%, 50\%, 70\%\}$ and the results are shown in Table 1. The experimental results illustrate that our method CASEN consistently achieves better performance than the baseline methods on each dataset. Specifically, for missing rate $p = 0.3$, compared to the second best method, CASEN improves ACC by 2.37%, 6.39%, 2.77%, 8.08% on BBC, Caltech20, Wikipedia and MNIST datasets, respectively, which demonstrates the effectiveness of CASEN. We would like to highlight several aspects of the experimental results: 1) Multi-view clustering methods generally obtain better performance than BSV, which indicates that different views can complement each other and leveraging multi-view complementary information is beneficial to improve clustering performance. 2) MIC, OMVC and DAIMC are based on matrix factorization. However, their clustering performance are limited because the local structure information is ignored and the learned latent representation cannot well capture data correlations. 3) UEAF and PIC leverage local structure for multi-view clustering, while they neglect to use global structures so they cannot obtain effective clustering results. 4) The proposed method CASEN performs better than the other methods by utilizing both local and global structure to capture comprehensive data correlations. Moreover, multi-view clustering and network training are jointly conducted so that the two tasks can promote each other to achieve better clustering performance.

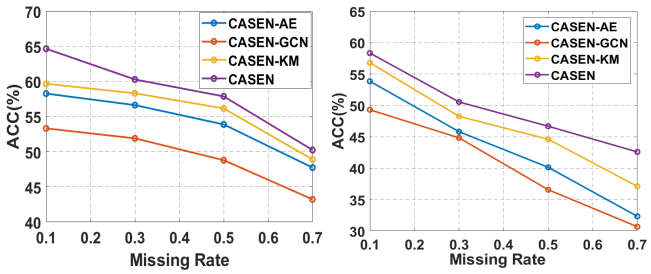
3.3 Component and Parameter Analysis

We study the effectiveness of each module of CASEN on Caltech20 and Wikipedia datasets. Three baseline methods are introduced: 1) CASEN-AE: Remove adaptive multi-view graph structure extraction module from CASEN, where only global structure is leveraged in the model. 2) CASEN-GCN: Remove multi-view autoencoder module from CASEN, where only local structure is used. 3) CASEN-KM: Perform k-means clustering on R to obtain clustering results instead of using multiple kernel clustering. From Figure 2 we can observe that by jointly exploiting global and local structure and conducting multi-view clustering through MKC, CASEN outperforms the other three degradation models, which verifies the effectiveness of each module.

The sensitivity analysis experiments are conducted on Caltech20 and Wikipedia datasets for $p = 0.3$ in Figure 3, where the two important parameters η_1 and η_2 in (8) are studied. We search η_1 and η_2 in $[10^{-4}, 10^{1}]$ and present how the clustering performance changes. It can be observed that the performance of our method is relatively stable with the two parameters. Promising performance can be obtained in a wide range for $\eta_1 \in [10^{-3}, 10^{-1}]$ and $\eta_2 \in [10^{-3}, 10^{0}]$.

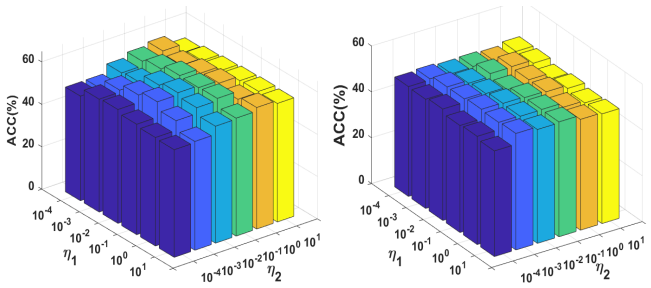
4 Conclusion

We propose an end-to-end trainable network CASEN for joint adaptive structure enhancing and incomplete multi-view clustering. Unlike the existing incomplete multi-view clustering methods that only use the incomplete multi-view structure information, CASEN further improves clustering performance by completing the missing features and integrating both global and local structures of multi-view data. Multiple kernel clustering is introduced to obtain reliable and accurate clustering results, and the clustering results in turn are used to guide network training through a self-supervision strategy. Extensive experiments conducted on several benchmark datasets demonstrate the effectiveness and reasonableness of CASEN and its advantages over the other methods.



(a) Caltech20 (b) Wikipedia

Figure 2: Component analysis experiments of CASEN on Caltech20 and Wikipedia datasets.



(a) Caltech20 (b) Wikipedia

Figure 3: Clustering performance of CASEN for different η_1 and η_2 on Caltech20 and Wikipedia datasets. The missing rate is 0.3.

Acknowledgments

This work was supported by National Key R&D Program of China (2018YFB1402600), and by the National Natural Science Foundation of China (61802028, 61772083, 61877006, 62002027), and sponsored by CCF-Baidu Open Fund, and by Beijing Nova Program (No. Z201100006820074).

References

[Cai *et al.*, 2013] Xiao Cai, Feiping Nie, Weidong Cai, and Heng Huang. Heterogeneous image features integration via multi-modal semi-supervised learning model. In *ICCV*, pages 1737–1744, 2013.

[Greene and Cunningham, 2006] Derek Greene and Pádraig Cunningham. Practical solutions to the problem of diag-

onal dominance in kernel document clustering. In *ICML*, pages 377–384, 2006.

[Hu and Chen, 2019a] Menglei Hu and Songcan Chen. Doubly aligned incomplete multi-view clustering. *arXiv preprint arXiv:1903.02785*, 2019.

[Hu and Chen, 2019b] Menglei Hu and Songcan Chen. One-pass incomplete multi-view clustering. In *AAAI*, volume 33, pages 3838–3845, 2019.

[Huang *et al.*, 2020] Zhenyu Huang, Peng Hu, Joey Tianyi Zhou, Jiancheng Lv, and Xi Peng. Partially view-aligned clustering. In *Advances in Neural Information Processing Systems*, volume 33, pages 2892–2902, 2020.

[Jiang *et al.*, 2019] Bo Jiang, Ziyan Zhang, Doudou Lin, Jin Tang, and Bin Luo. Semi-supervised learning with graph learning-convolutional networks. In *CVPR*, pages 11313–11320, 2019.

[Kang *et al.*, 2020] Zhao Kang, Xinjia Zhao, Shi, chong Peng, Hongyuan Zhu, Joey Tianyi Zhou, Xi Peng, Wenyu Chen, and Zenglin Xu. Partition level multiview subspace clustering. *Neural Networks*, 122:279–288, 2020.

[Kang *et al.*, 2021] Zhao Kang, Zhiping Lin, Xiaofeng Zhu, and Wenbo Xu. Structured graph learning for scalable subspace clustering: From single-view to multi-view. *IEEE Transactions on Cybernetics*, 2021.

[LeCun *et al.*, 1998] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[Li *et al.*, 2014] Shao-Yuan Li, Yuan Jiang, and Zhi-Hua Zhou. Partial multi-view clustering. In *AAAI*, volume 28, 2014.

[Li *et al.*, 2020] Xuelong Li, Han Zhang, Rong Wang, and Feiping Nie. Multi-view clustering: A scalable and parameter-free bipartite graph fusion method. *IEEE TPAMI*, 2020.

[Liu *et al.*, 2019] Xinwang Liu, Xinzhong Zhu, Miaomiao Li, Lei Wang, En Zhu, Tongliang Liu, Marius Kloft, Dinggang Shen, Jianping Yin, and Wen Gao. Multiple kernel k k-means with incomplete kernels. *IEEE TPAMI*, 42(5):1191–1204, 2019.

[Munkres, 1957] James Munkres. Algorithms for the assignment and transportation problems. *Journal of the society for industrial and applied mathematics*, 5(1):32–38, 1957.

[Peng *et al.*, 2019] Xi Peng, Zhenyu Huang, Jiancheng Lv, Hongyuan Zhu, and Joey Tianyi Zhou. COMIC: Multi-view clustering without parameter selection. In *ICML*, volume 97 of *Proceedings of Machine Learning Research*, pages 5092–5101. PMLR, 09–15 Jun 2019.

[Rasiwasia *et al.*, 2010] N. Rasiwasia, J. Costa Pereira, E. Coviello, G. Doyle, G.R.G. Lanckriet, R. Levy, and N. Vasconcelos. A New Approach to Cross-Modal Multimedia Retrieval. In *ACM Multimedia*, pages 251–260, 2010.

[Shao *et al.*, 2015] Weixiang Shao, Lifang He, and Philip S. Yu. Multiple incomplete views clustering via weighted nonnegative matrix factorization with $l_{2,1}$ regularization. In *ECML PKDD*, pages 318–334. Springer, 2015.

[Shao *et al.*, 2016] Weixiang Shao, Lifang He, Chun-Ta Lu, and Philip S. Yu. Online multi-view clustering with incomplete views. In *2016 IEEE International Conference on Big Data (Big Data)*, pages 1012–1017. IEEE, 2016.

[Tzortzis and Likas, 2012] Giorgos Tzortzis and Aristidis Likas. Kernel-based weighted multi-view clustering. In *ICDM*, pages 675–684, 2012.

[Wang *et al.*, 2018] Qianqian Wang, Zhengming Ding, Zhiqiang Tao, Quanxue Gao, and Yun Fu. Partial multi-view clustering via consistent gan. In *ICDM*, pages 1290–1295. IEEE, 2018.

[Wang *et al.*, 2019] Hao Wang, Linlin Zong, Bing Liu, Yan Yang, and Wei Zhou. Spectral perturbation meets incomplete multi-view data. *arXiv preprint arXiv:1906.00098*, 2019.

[Wen *et al.*, 2016] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *ECCV*, volume 9911, pages 499–515. Springer, 2016.

[Wen *et al.*, 2019] Jie Wen, Zheng Zhang, Yong Xu, Bob Zhang, Lunke Fei, and Hong Liu. Unified embedding alignment with missing views inferring for incomplete multi-view clustering. In *AAAI*, pages 5393–5400, 2019.

[Wen *et al.*, 2020a] Jie Wen, Zheng Zhang, Yong Xu, Bob Zhang, Lunke Fei, and Guo-Sen Xie. Cdimc-net: Cognitive deep incomplete multiview clustering network. In *IJCAI*, pages 3230–3236, 2020.

[Wen *et al.*, 2020b] Jie Wen, Zheng Zhang, Zhao Zhang, Zhihao Wu, Lunke Fei, Yong Xu, and Bob Zhang. Dimc-net: Deep incomplete multi-view clustering network. In *ACM Multimedia*, pages 3753–3761, 2020.

[Xu *et al.*, 2019] Cai Xu, Ziyu Guan, Wei Zhao, Hongchang Wu, Yunfei Niu, and Beilei Ling. Adversarial incomplete multi-view clustering. In *IJCAI*, pages 3933–3939, 2019.

[Yang *et al.*, 2021] Mouxing Yang, Yunfan Li, Zhenyu Huang, Zitao Liu, Peng Hu, and Xi Peng. Partially view-aligned representation learning with noise-robust contrastive loss. In *CVPR*, June 2021.

[Zhang *et al.*, 2020a] Changqing Zhang, Yajie Cui, Zongbo Han, Joey Tianyi Zhou, Huazhu Fu, and Qinghua Hu. Deep partial multi-view learning. *IEEE TPAMI*, 2020.

[Zhang *et al.*, 2020b] Changqing Zhang, Huazhu Fu, Qinghua Hu, Xiaochun Cao, Yuan Xie, Dacheng Tao, and Dong Xu. Generalized latent multi-view subspace clustering. *IEEE TPAMI*, 42(1):86–99, 2020.

[Zhao *et al.*, 2016] Handong Zhao, Hongfu Liu, and Yun Fu. Incomplete multi-modal visual data grouping. In *IJCAI*, pages 2392–2398, 2016.

[Zhou *et al.*, 2019] Wei Zhou, Hao Wang, and Yan Yang. Consensus graph learning for incomplete multi-view clustering. In *PAKDD*, pages 529–540. Springer, 2019.