Letter

Contrastive Consensus Graph Learning for Multi-View Clustering

Shiping Wang, Xincan Lin, Zihan Fang, Shide Du, and Guobao Xiao

Dear Editor,

This letter proposes a contrastive consensus graph learning model for multi-view clustering. Graphs are usually built to outline the correlation between multi-model objects in clustering task, and multi-view graph clustering aims to learn a consensus graph that integrates the spatial property of each view. Nevertheless, most graph-based models merely consider the overall structure from all views but neglect the local spatial consistency between diverse views, resulting in the lack of global spatial consistency in the learned graph. To overcome this issue, a deep convolutional network is built to explore latent local spatial information from raw affinity graphs. Specifically, we employ a consensus graph constraint to preserve the global consistency between the learned graph and raw graphs. Furthermore, a contrastive reconstruction loss is introduced to achieve the sample-level approximation between reconstructed graphs and raw graphs, which facilitates the network to enhance the consensus graph learning. Experiments on six classical datasets demonstrate that the proposed model outperforms other nine state-of-the-art algorithms.

Related work: In real-world applications, multimedia data are usually generated from multiple ways and presented in diverse forms, referred as multi-view data. Compared with single-view data, multi-view data contains more comprehensive information, which makes multi-view learning be a hot spot. Previous work [1], [2] has been devoted to this field and achieves satisfactory results. As an important branch of multi-view learning, multi-view clustering aims to effectively fuse information and discover the underlying clustering structure shared by diverse views. Since each view has a distinct focus on the same object, multi-view data tends to be complementary and consistent. Therefore, it is critical to integrate multi-view features and fully exploit the consistency and complementarity to obtain shared discriminative representations.

Plenty of research has attempted to extract shared information from multi-view data to improve clustering performance [3]–[5], among which the graph-based approach is a mainstream issue. Graphs are typically built to represent relationships between different objects, with nodes corresponding to data objects and weighted edges depicting the similarity between data points. Generally, multi-view graph clustering methods can be roughly boiled down to two stages: first learning a consensus graph from all views, then applying post-processing techniques on the learned graph to obtain clustering results [6]. Since the quality of the learned graph can directly determine the clustering effect, how to learn a high-quality graph becomes a critical issue [7]. For that, [3] proposes a self-weighted method to explore a Laplacian constrained graph and directly obtain the clustering result without any follow-up processing. Reference [8] designs a regularization term to adaptively learn weights of the views for diversity enrichment and redundancy reduction. Furthermore, to avoid the effect of the predefined graph quality, [9] introduces a disagreement cost function and constrains the rank of the Laplacian matrix of the learned graph. However, these methods merely focus on optimal weight learning for each view and neglect the local spatial consistency between different views, resulting in the lack of spatial integrity.

Besides, various deep clustering methods are constructed to exploit latent semantic information among data. For instance, [10] proposes a deep canonical correlation analysis framework, which employs two deep neural networks to extract implicit features of each view. To better explore complementary information, [11] presents a semi-non-negative matrix factorization method for learning hierarchical semantics of multi-view data. Reference [12] integrates the within-view invariance, the between-view consistency, and the nonlinear embedding network to learn a common space for spectral clustering. Recently, [13] proposes an instance-level and cluster-level contrastive learning method for clustering and [14] lifts the instance-level consistency to the cluster-level consistency for graph learning. Furthermore, [15] learns an informative and consistent representation by maximizing the mutual information between diverse views by introducing contrastive learning. Despite these autoencoder-based models can effectively extract latent information, they solely achieve the element-level reconstructed approximation but lack of sample-level approximation, which are not conducive to the consensus graph learning.

Based on the above observations, we propose a multi-view clustering network by utilizing a convolutional autoencoder for learning a consensus graph. The proposed network is illustrated in Fig. 1, which is composed of a graph construction layer and a symmetric convolutional autoencoder. Specifically, we integrate convolutional autoencoder, consensus graph learning, and contrastive reconstruction learning into a unified framework to obtain a common graph with spatial consistency. The main contributions are summarized as: 1) Build a convolutional autoencoder to capture the local spatial information from different views and obtain a latent consensus graph; 2) A consensus graph loss is proposed to approximate the consensus graph with all raw graphs so as to preserve the global spatial consistency of the learned graph; 3) Introduce a contrastive reconstruction loss to constrain the sample-level consistency, and to enhance the similarity between reconstructed graphs and raw graphs.

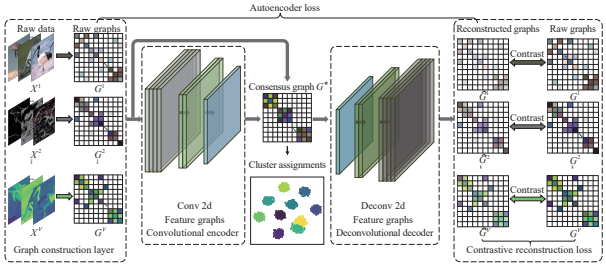

Fig. 1. A scheme of the proposed model, which consists of a graph construction layer and a convolutional autoencoder network. Given raw data, the proposed model first constructs affinity graphs by a specific graph construction method. Accordingly, the affinity graphs are fed to the convolutional autoencoder to learn a consensus graph by jointly minimizing the autoencoder loss, the consensus graph loss and the contrastive reconstruction loss.

Contrastive consensus graph learning for multi-view clustering: Given a multi-view dataset of V views $\mathcal{X} = \{\mathbf{X}^v\}_{v=1}^V$, where $\mathbf{X}^v \in \mathbb{R}^{N\times D_v}$ is N instances with D_v dimension from v-th view, and the affinity graphs are denoted by $\mathcal{G} = \{\mathbf{G}^v\}_{v=1}^V$, where $\mathbf{G}^v \in \mathbb{R}^{N\times N}$ is the affinity matrix from v-th view. Suppose that $\mathbf{G}^*$ is the learned consensus graph, and the reconstructed affinity graphs are represented by $\hat{\mathcal{G}} = \{\hat{\mathbf{G}}^v\}_{v=1}^V$. Besides, $\otimes$ and $\circledast$ are utilized to denote convolution and

Corresponding author: Guobao Xiao.

Citation: S. P. Wang, X. C. Lin, Z. H. Fang, S. D. Du, and G. B. Xiao, "Contrastive consensus graph learning for multi-view clustering," *IEEE/CAA J. Autom. Sinica*, vol. 9, no. 11, pp. 2027–2030, Nov. 2022.

S. P. Wang, X. C. Lin, Z. H. Fang, and S. D. Du are with the College of Computer and Data Science, Fuzhou University, Fuzhou 350108, China (e-mail: shipingwangphd@163.com; xincanlinms@gmail.com; Fzihan11@163.com; dushidems@gmail.com).

G. B. Xiao is with the College of Computer and Control Engineering, Minjiang University, Fuzhou 350108, China (e-mail: x-gb@163.com).

Digital Object Identifier 10.1109/JAS.2022.105959

deconvolution operation, respectively. The architecture of the proposed model is described as follows.

1) Graph construction layer: The nearest neighbors method is first utilized to generate the adjacency matrices of raw data, then the affinity matrices are constructed by Gaussian kernel function as

$$\mathbf{G}_{ij}^{v}=\begin{cases}\exp\left(-\dfrac{\left\|\mathbf{X}_i^v-\mathbf{X}_j^v\right\|_2^2}{2\varepsilon^2}\right), & \mathbf{X}_i^v,\mathbf{X}_j^v \text{ are connected}\\ 0, & \text{otherwise}\end{cases} \tag{1}$$

where $\mathbf{G}_{ij}^{v}$ is the similarity between the i-th and the j-th instance in the v-th view, and ε is the kernel bandwidth that controls the radial range of the Gaussian function.

2) Convolutional autoencoder: As each graph can be taken as a 2d grid in which each point implies correlation between nodes, and 2d convolution kernel is capable of mining neighborhood information through local receptive fields in a 2d space, the local spatial association between instances can be explored in the encoder. Therefore, the convolutional autoencoder is built by multiple 2d convolutional layers as the encoder and multiple 2d deconvolutional layers as the decoder. Assume that $\mathbf{A}\in\mathbb{R}^{s\times s}$ is a 2d convolution kernel of size $s\times s$. With the window sliding mechanism, $\mathbf{A}$ enables to dig out the spatial information in a local receptive field of $s\times s$ on all graphs, and an enhanced feature graph with specific local property can be obtained. Through multiple convolution kernels formed by $\mathbf{A}$, different local features are continuously extracted and explored. Simultaneously, the iterative operation of multiple convolutional layers discover deep latent information and further learn a consistent graph with a relatively complete spatial structure. The implicit feature graphs in the encoder are formalized as

$$\tilde{\mathbf{G}}_i^k=\begin{cases}\sigma\left(\displaystyle\sum_{j=1}^{V}\mathbf{G}^j\otimes\bar{W}_i^k+\bar{b}_i^k\right), & i=1\\ \sigma\left(\displaystyle\sum_{j=1}^{K_{i-1}}\tilde{\mathbf{G}}_{i-1}^j\otimes\bar{W}_i^k+\bar{b}_i^k\right), & i=2,3,\ldots,L\end{cases} \tag{2}$$

where $\sigma(\cdot)$ denotes an activation function, $\bar{K}_i$ are the number of convolution kernels in the i-th layer, $\bar{W}_i^k$ and $\bar{b}_i^k$ are the weight parameters and the bias term of the k-th convolution kernel of the i-th layer, and $\bar{L}$ is the number of convolutional layers. The output of the encoder is the learned consensus graph, i.e., $\mathbf{G}^*=\tilde{\mathbf{G}}_{\bar{L}}^1$. Due to the inverse of convolution, deconvolution seeks to reconstruct approximate outputs of inputs from encodings. Therein, the reconstructed feature graphs in the decoder can be obtained by

$$\hat{\mathbf{G}}_i^k=\begin{cases}\sigma\left(\mathbf{G}^*\circledast\hat{W}_i^k+\hat{b}_i^k\right), & i=1\\ \sigma\left(\sum_{j=1}^{\hat{K}_{i-1}}\hat{\mathbf{G}}_{i-1}^j\circledast\hat{W}_i^k+\hat{b}_i^k\right), & i=2,3,\ldots,\hat{L}\end{cases} \tag{3}$$

where $\hat{W}_i^k$, $\hat{b}_i^k$, $\hat{K}_i$ and $\hat{L}$ are synonymous with those mentioned in the encoder. Through multiple deconvolutional layers, the outputs of the decoder are the reconstructed graphs $\hat{\mathcal{G}}$, i.e., $\hat{\mathcal{G}}=\{\hat{\mathbf{G}}_{\hat{L}}^k\}_{k=1}^{\hat{K}_{\hat{L}}}$.

Objective function:

1) Autoencoder loss $\mathcal{L}_{ae}$: The autoencoder is to extract latent information motivated by preserving the reconstructions as similar as possible to the inputs, which can be taken as an element-level approximation between reconstructions and inputs. The mean square error is utilized to measure the autoencoder loss, calculated by

$$\mathcal{L}_{ae}=\sum_{v=1}^{V}\left\|\hat{\mathbf{G}}^v-\mathbf{G}^v\right\|_F^2. \tag{4}$$

2) Consensus graph loss $\mathcal{L}_z$: Though the graphs of diverse views differ from each other in the local spatial scope, they essentially admit the consistent global spatial structure. With this assumption, the convolutional autoencoder is expected to learn a spatial consistent graph from different views. Therein, the consensus graph loss is proposed to hold the global spatial consistency, defined as

$$\mathcal{L}_z=\sum_{v=1}^{V}\omega_v^{(p)}\left\|\mathbf{G}^*-\mathbf{G}^v\right\|_F^2 \tag{5}$$

where $\omega_v^{(p)}$ is a weight parameter to trade-off the contributions of graphs from diverse views to the consensus graph, with exponent p to control the impact of the weight, satisfying $\sum_{v=1}^{V}\omega_v=1$ to normalize the weights of diverse perspectives. Intuitively, the solution of ω_v is the pole value while minimizing $\mathcal{L}_z$. Following Lagrangian multiplier method, the Lagrange function of $\mathcal{L}_z$ is first obtained by:

$$\sum_{v=1}^{V}\omega_v^{(p)}\left\|\mathbf{G}^*-\mathbf{G}^v\right\|_F^2+\sum_{v=1}^{V}\lambda_v\omega_v-1 \tag{6}$$

where λ_v is a Lagrange multiplier for the constraint of ω_v. Next, taking the partial derivative with respect to ω_v of (6) and setting the derivative to be zero, we know

$$\frac{\partial\omega_v^{(p)}\|\mathbf{G}^*-\mathbf{G}^v\|_F^2}{\partial\omega_v}+\frac{\partial\lambda_v\omega_v}{\partial\omega_v}=0. \tag{7}$$

With (7), ω_v can be obtained with λ_v and then substituted to $\sum_{v=1}^{V}\omega_v=1$ for calculating λ_v, finally ω_v can be gained as

$$\omega_v=\frac{\left(\|\mathbf{G}^*-\mathbf{G}^v\|_F^2\right)^{\frac{1}{1-p}}}{\sum_{v=1}^{V}\left(\|\mathbf{G}^*-\mathbf{G}^v\|_F^2\right)^{\frac{1}{1-p}}}. \tag{8}$$

3) Contrastive reconstruction loss $\mathcal{L}_c$: The autoencoder not only serves as the function of capturing latent information, but also achieves the reconstruction of the intermediate representation. Consequently, the quality of the reconstructed graphs produces a certain impact on the merit of the learned consensus graph. However, the autoencoder loss $\mathcal{L}_{ae}$ merely realizes an element-level approximation but ignores the sample-level approximation, resulting in the incomplete similarity between the reconstructed graphs and raw graphs, which is detrimental to consensus graph learning. To remedy this deficiency, the contrastive reconstruction loss $\mathcal{L}_c$ is introduced.

Since the decoding outputs are constructed by a functional mapping of the inputs, the reconstructed graphs can be regarded as augmentations of original graphs. In v-th view, the reconstructed graph $\hat{\mathbf{G}}^v$ is an augmentation of $\mathbf{G}^v$, then we obtain $2N$ similarity samples $\{\hat{\mathbf{G}}_i^v,\mathbf{G}_i^v\}_{v=1}^{V}$. For the i-th reconstructed sample $\hat{\mathbf{G}}_i^v$, the corresponding raw sample $\mathbf{G}_i^v$ is chosen as a positive pair $\{\hat{\mathbf{G}}_i^v,\mathbf{G}_i^v\}$, and the other $2N-2$ pairs are regarded as negative ones. To make samples distinguishable, the contrastive loss for $\hat{\mathbf{G}}_i^v$ is gained by

$$\ell(\hat{\mathbf{G}}_i^v)=-\log\frac{\exp\left(\dfrac{s(\hat{\mathbf{G}}_i^v,\mathbf{G}_i^v)}{\tau}\right)}{\sum_{j=1}^{N}\left[\exp\left(\dfrac{s\left(\hat{\mathbf{G}}_i^v,\hat{\mathbf{G}}_j^v\right)}{\tau}\right)+\exp\left(\dfrac{s\left(\hat{\mathbf{G}}_i^v,\mathbf{G}_j^v\right)}{\tau}\right)\right]} \tag{9}$$

where τ is a temperature parameter to control the smoothness, and $s(\hat{\mathbf{G}}_i^v,\mathbf{G}_i^v)$ denotes the cosine similarity between $\hat{\mathbf{G}}_i^v$ and $\mathbf{G}_i^v$ calculated by $s(\hat{\mathbf{G}}_i^v,\mathbf{G}_i^v)=\hat{\mathbf{G}}_i^v(\mathbf{G}_i^v)^T/\left(\|\hat{\mathbf{G}}_i^v\|\|\mathbf{G}_i^v\|\right)$. Hence, the contrastive loss for v-th view is calculated by the average contrastive loss of all positive samples, and the contrastive loss of all views are accumulated to form the contrastive reconstruction loss $\mathcal{L}_c$ as

$$\mathcal{L}_c=\frac{1}{2N}\sum_{v=1}^{V}\sum_{i=1}^{N}\left(\ell(\hat{\mathbf{G}}_i^v)+\ell\left(\mathbf{G}_i^v\right)\right). \tag{10}$$

With all the losses as introduced above, the total objective function $\mathcal{L}$ is denoted as

$$\mathcal{L}=\mathcal{L}_{ae}+\alpha\mathcal{L}_z+\beta\mathcal{L}_c \tag{11}$$

where α and β are employed to balance the impact of consensus graph loss and the contrastive reconstruction loss. In the process of minimizing the total loss, the network is steered to learn a consensus graph as summarized in Algorithm 1. Finally, the clustering result is obtained by conducting spectral clustering on the learned graph.

Algorithm 1 Contrastive Consensus Graph Learning

Input: Multi-view data $\mathcal{X}$, the number of nearest neighbors k, learning rate lr, training epochs t, weight parameters α and β.
Output: Consensus graph $\mathbf{G}^*$.
1: Generate adjacency graphs by KNN from $\mathcal{X}$, and then construct affinity graphs $\mathcal{G}$ using (1).
2: Initial the network weights by Xavier normalization.
3: **for** $epoch = 1 \rightarrow t$ **do**
4: Calculate feature graphs $\mathcal{G}$ and $\hat{\mathcal{G}}$ by (2) and (3).
5: Compute the autoencoder loss $\mathcal{L}_{ae}$ by (4).
6: Obtain the contrastive recounstruction loss $\mathcal{L}_c$ by (9) and (10).
7: Calculate ω_i for each view through (8), then compute the consensus graph loss $\mathcal{L}_z$ by (5).
8: Calculate the total loss $\mathcal{L}$ by (11).
9: Update network parameters by gradient descent method.
10: **end for**
11: **return** Ouput of the encoder $\mathbf{G}^*$.

The total time complexity mainly comes from the construction time of K nearest neighbor (KNN) matrices and the forward computation of the convolutional autoencoder. The time spent on graph construction sorting and computing similarity, and they cost $O(VN\log N)$ and $O(\sum_{v=1}^{V} D_v N^2)$, respectively. Thus, the time complexity of graph construction is $O(VN\log N+\sum_{v=1}^{V} D_v N^2)$. As the size of feature graphs and convolution kernel are $N\times N$ and $s\times s$, the forward computing time consumes $O(\sum_{l=1}^{2L} N^2\times s_l^2 c_{l-1}c_l)$, where c denotes the convolutional channels. Therein, the overall time complexity requires $O(VN\log N+(\sum_{v=1}^{V} D_v+\sum_{l=1}^{2L} s_l^2 c_{l-1}c_l)N^2)$, approximated to $O(N^2)$. Besides, the space complexity is determined by the weight parameters of the convolution kernels and feature graphs, which costs $O(\sum_{l=1}^{2L}(s_l^2 c_{l-1}c_l + c_l N^2))$ and can be approximated as $O(N^2)$.

Experiments:

1) Datasets: The experiments are conducted on six classical datasets, and a brief description is illustrated as follows. Specifically, ALOI contains 1079 object images with four color features. Hand-Written (HW) are 2000 handwritten digits images with six views. Caltech101-20 is a object recognition dataset with 101 categories, and we select 2386 samples of 20 classes for testing. Youtube consists of 2000 samples including three visual features and three audio features. NUS-WIDE is comprised of 1600 web images of six available features. MNIST10k is a image dataset of 10000 handwritten digits with IsoProjection, linear discriminant analysis (LDA) and neighborhood preserving embedding (NPE) features as three views.

2) Comparisons and parameter settings: The proposed method is compared with the following nine methods. Best single view (BSV) is adopted to record the best spectral single-view clustering performance of all raw affinity graphs. The rest compared clustering methods are tensorized multi-view subspace representation learning (TMSRL) [16], multi-view clustering via deep matrix factorization (DMF-MVC) [11], deep generalized canonical correlation analysis (DGCCA) [10], multi-veiw spectral clustering network (MvSCN) [12], multiview consensus graph clustering (MCGC) [9], graph-based multi-view clustering (GMC) [7], binary multi-view clustering (BMVC) [17] and consensus graph learning (CGL) [6]. All the parameters involved in compared algorithms are set to the recommended values in their papers. For the proposed model, there are both 3 convolutional and deconvolutional layers and the size of all convolution kernels is 3×3, where each layer is followed by a ReLU activation. Strides of horizontal and vertical directions are both 1 with one zero-padding layer to obtain feature graphs with the size of $n\times n$. In the encoder, the number of convolution kernels are [4, 2, 1] in order, while there are [2, 4, V] deconvolution kernels in the decoder. For all datasets, the number of nearest neighbors k is chosen as 10 with the bandwidth $\varepsilon = 1$, the exponent p of consensus weight is set to -1, and the temperature τ in the contrastive loss is fixed to 1, and we set $\alpha = 1$ and $\beta = 1$ as default. Moreover, Adam optimizer is utilized to accelerate the minimization of the total loss with $lr = 0.01$. Uniformly, we train 300 epochs on all datasets to obtain consensus graphs. All experiments are conducted for 10 times, then the mean and the standard deviation of the clustering performance are computed as the final results.

3) Clustering results: The classical metric accuracy (ACC) is adopted to evaluate the clustering performance. The clustering results of all compared algorithms are presented in Table 1, where we can obtain the following observations. Compared with BSV, the proposed model exhibits better performance. Compared with the tensor-based method TMSRL, the proposed model exhibits significant clustering superiority on all datasets. As for deep methods such as DGCCA, our method also gains superior performance on most datasets, which demonstrates the effectiveness of the convolutional autoencoder. Furthermore, compared with graph-based methods, the proposed model still obtains higher accuracy especially on HW and MNIST where the mean accuracy is close to 100%. In summary, the proposed model is capable of learning a well clustered graph and achieving satisfactory clustering results.

Table 1. The ACC (MEAN±STD%) of Multi-View Clustering Algorithms, Where the Best and Second Best Results are Marked in Red and Blue Respectively, And "–" Denotes the Failed Results.

Methods\Datasets	ALOI	Caltech101-20	HW	MNIST10k	NUS-WIDE	Youtube
BSV	86.55±0.00	**65.58±0.00**	63.50±0.29	90.30±0.00	30.73±0.34	34.50±0.87
TMSRL	61.94±0.19	50.60±5.01	85.74±0.08	–	33.01±0.05	29.82±0.70
DMF-MVC	79.52±0.00	54.81±0.51	34.81±0.18	19.82±0.02	33.01±0.05	28.13±1.17
DGCCA	57.31±0.00	63.45±0.91	64.10±2.80	29.50±0.09	27.50±1.10	29.40±0.20
MvSCN	56.00±2.50	38.30±0.75	50.30±1.00	73.27±6.23	30.20±1.40	24.40±0.20
MCGC	55.51±0.00	62.91±0.00	95.50±0.00	61.03±0.00	21.56±0.00	30.00±0.00
BMVC	59.59±0.00	47.41±0.64	85.63±0.57	53.55±0.00	**36.64±1.32**	**46.53±0.71**
GMC	64.87±0.00	45.64±0.00	85.90±0.29	**90.37±0.00**	20.06±0.00	11.65±0.00
CGL	**92.12±0.00**	54.82±0.00	**97.40±0.00**	–	31.37±0.00	34.50±0.87
Ours	**93.51±0.00**	**81.32±0.83**	**99.50±0.04**	**98.33±0.00**	**59.99±0.97**	**47.77±2.72**

4) Ablation study: As shown in Table 2, the ablation study is performed to investigate the influence of the consensus graph loss and the contrastive loss. The results indicate that the proposed model performs poorly with only autoencoder loss. After adding the consensus graph loss, the clustering performance of the proposed model is significantly improved. Further introducing the contrastive reconstruction loss, the proposed model performs best. It can be inferred that the consensus graph constraint can guide the network to effectively explore the discriminative spatial information from diverse views. Simultaneously, with the contrastive reconstruction loss, the sample-level similarity between the reconstructed graphs and raw graphs can be strengthened, and in turn enhancing the graph learning.

Table 2. The Ablation Study on ALOI, HW and Youtube w.r.t ACC (Mean±STD%), where the Best Results are in Bold.

Loss	ALOI	HW	Youtube
$\mathcal{L}_{ae}$	82.47±1.26	85.74±0.00	17.40±0.22
$\mathcal{L}_{ae}+\mathcal{L}_z$	85.69±3.18	96.99±0.00	37.59±0.84
$\mathcal{L}_{ae}+\mathcal{L}_z+\mathcal{L}_c$	**93.51±0.00**	**99.50±0.00**	**47.77±2.72**

5) Convergence and parameter sensitivity: The objective function values with the number of epochs are illustrated in Fig. 2(a), where MNIST10k is scaled by 10 times to keep all curves in the same inter-

val. It can be seen that the objective function value decreases rapidly and stabilizes after 250 epochs on all datasets, indicating that the proposed model can converge to a stable value. In addition, parameter sensitivity experiments are conducted to investigate the influence of α and β and the results are presented in Fig. 2(b), with α and β ranging in $\{10^{-4}, 10^{-3}, \ldots, 10^{4}\}$. It can be observed that the proposed model performs well when both α and β are in the same or similar order of magnitude, suggesting that the consensus graph constrain and the contrastive reconstruction constrain play a similar importance to the consensus graph learning.

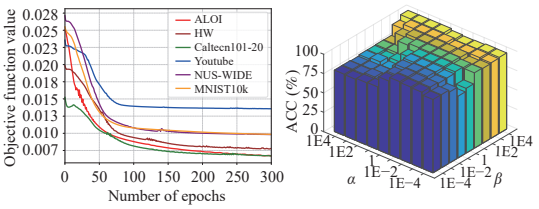

Fig. 2. Convergence and parameter sensitivity of the proposed method. (a) Curves of objective function values with the number of epochs; (b) Clustering metric ACC on HW with varied α and β.

Conclusions: This letter proposed a contrastive consensus graph learning model to learn a consensus graph, which adopted a convolutional autoencoder network to efficiently explore the latent spatial association among data. With the constraints of the consensus graph loss, the learned graph was able to maintain global spatial consistency across diverse views. Furthermore, a contrastive reconstruction loss was introduced to achieve sample-level approximations between the reconstructed graphs and the raw graphs, further to enhance the consistency of the learned graph. Experimental results demonstrated the superiority of the proposed model.

Acknowledgments: This work was supported by the National Natural Science Foundation of China (U21A20472, 62072223), the National Key Research and Development Plan of China (2021YFB 3600503), and the Natural Science Foundation of Fujian Province (2020J01130193, 2020J01131199).

References

[1] T. Zhou, M. Chen, and J. Zou, " Reinforcement learning based data fusion method for multi-sensors," *IEEE/CAA J. Autom. Sinica* , vol. 7, no. 6, pp. 1489–1497, 2020.

[2] Y. Wang, Z. Zhang, and Y. Lin, "Multi-cluster feature selection based on isometric mapping," *IEEE/CAA J. Autom. Sinica*, vol. 9, no. 3, pp. 570–572, 2021.

[3] F. Nie, J. Li, and X. Li, " Self-weighted multiview clustering with multiple graphs," in *Proc. Int. Joint Conf. Artificial Intelligence*, 2017, pp. 2564–2570.

[4] S. Wang, Z. Chen, S. Du, and Z. Lin, "Learning deep sparse regularizers with applications to multi-view clustering and semi-supervised classification," *IEEE Trans. Pattern Analysis and Machine Intelligence*, pp. 1–14, 2021, DOI: 10.1109/TPAMI.2021.3082632.

[5] S. Du, Z. Liu, Z. Chen, W. Yang, and S. Wang, " Differentiable bi-sparse multi-view co-clustering," *IEEE Trans. Signal Processing* , vol. 69, pp. 4623–4636, 2021.

[6] Z. Li, C. Tang, X. Liu, X. Zheng, G. Yue, W. Zhang, and E. Zhu, "Consensus graph learning for multi-view clustering," *IEEE Trans. Multimedia*, pp. 1–12, 2021, DOI: 10.1109/TMM.2021.3081930.

[7] H. Wang, Y. Yang, and B. Liu, " GMC: Graph-based multi-view clustering," *IEEE Trans. Knowledge and Data Engineering*, vol. 32, no. 6, pp. 1116–1129, 2019.

[8] C. Tang, X. Zhu, X. Liu, M. Li, Wa ng, C. Zhang, and L. Wang, "Learning a joint affinity graph for multiview subspace clustering," *IEEE Trans. Multimedia*, vol. 21, no. 7, pp. 1724–1736, 2018.

[9] K. Zhan, F. Nie, J. Wang, and Y. Yang, "Multiview consensus graph clustering," *IEEE Trans. Image Processing* , vol. 28, no. 3, pp. 1261–1270, 2019.

[10] A. Benton, H. Khayrallah, B. Gujral, D. A. Reisinger, S. Zhang, and R. Arora, " Deep generalized canonical correlation analysis," in *Proc. Workshop Representation Learning for NLP*, 2019, pp. 1–6.

[11] H. Zhao, Z. Ding, and Y. Fu, "Multi-view clustering via deep matrix factorization," in *Proc. AAAI Conf. Artificial Intelligence*, 2017, pp. 2921–2927.

[12] Z. Huang, J. T. Zhou, X. Peng, C. Zhang, H. Zhu, and J. Lv, "Multi-view spectral clustering network," in *Proc. Int. Joint Conf. Artificial Intelligence*, 2019, pp. 2563–2569.

[13] Y. Li, P. Hu, Z. Liu, D. Peng, J. T. Zhou, and X. Peng, "Contrastive clustering," in *Proc. AAAI Conf. Artificial Intelligence*, 2021, pp. 8547–8555.

[14] H. Zhong, J. Wu, C. Chen, J. Huang, M. Deng, L. Nie, Z. Lin, and X.-S. Hua, " Graph contrastive clustering," in *Proc. IEEE/CVF Int. Conf. Computer Vision*, 2021, pp. 9224–9233.

[15] Y. Lin, Y. Gou, Z. Liu, B. Li, J. Lv, and X. Peng, "COMPLETER: Incomplete multi-view clustering via contrastive prediction," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2021, pp. 11174–11183.

[16] C. Zhang, H. Fu, J. Wang, W. Li, X. Cao, and Q. Hu, "Tensorized multi-view subspace representation learning," *Int. J. Computer Vision*, vol. 128, no. 8, pp. 2344–2361, 2020.

[17] Z. Zhang, L. Liu, F. Shen, H. T. Shen, and L. Shao, "Binary multi-view clustering," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 41, no. 7, pp. 1774–1782, 2019.