Enhancing Ensemble Clustering with Adaptive High-Order Topological Weights

Jiaxuan Xu[1], **Taiyong Li**[2*], **Lei Duan**[1*]

[1] School of Computer Science, Sichuan University, Chengdu, China
[2] School of Computing and Artificial Intelligence, Southwestern University of Finance and Economics, Chengdu, China
xujx@stu.scu.edu.cn, litaiyong@gmail.com, leiduan@scu.edu.cn

Abstract

Ensemble clustering learns more accurate consensus results from a set of weak base clustering results. This technique is more challenging than other clustering algorithms due to the base clustering result set's randomness and the inaccessibility of data features. Existing ensemble clustering methods rely on the Co-association (CA) matrix quality but lack the capability to handle missing connections in base clustering. Inspired by the neighborhood high-order and topological similarity theories, this paper proposes a topological ensemble model based on high-order information. Specifically, this paper compensates for missing connections by mining neighborhood high-order connection information in the CA matrix and learning optimal connections with adaptive weights. Afterward, the learned excellent connections are embedded into topology learning to capture the topology of the base clustering. Finally, we incorporate adaptive high-order connection representation and topology learning into a unified learning framework. To the best of our knowledge, this is the first ensemble clustering work based on topological similarity and high-order connectivity relations. Extensive experiments on multiple datasets demonstrate the effectiveness of the proposed method. The source code of the proposed approach is available at https://github.com/ltyong/awec.

Introduction

Clustering is an important type of unsupervised learning widely used in pattern recognition (Liu et al. 2017), medical data mining (Dai et al. 2022), object detection (Wang et al. 2020), and other fields (Chen et al. 2022). Unlike supervised learning, which has label information to refer to, accurate labels are transparent to clustering, making it more challenging. In general, clustering algorithms divide different clusters by extracting the internal associations of the data feature. Many representative clustering algorithms have been proposed, such as density peak clustering (Rodriguez and Laio 2014), multi-view clustering (Wang et al. 2018), and subspace clustering (Elhamifar and Vidal 2013; Liu et al. 2012). Nevertheless, the variety in data types and distributions compels researchers to devise clustering algorithms tailored to particular datasets. Ensemble clustering has received much attention in recent years and can contribute to solving this problem. Ensemble clustering first adopts any clustering algorithm to generate a set of base clustering result labels, and then applies the consensus function to obtain a more robust consensus result.

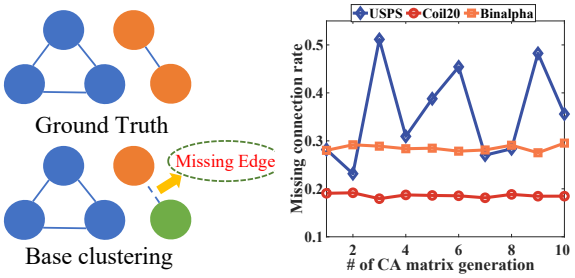


Figure 1: The missing connection refers to two samples of the same class in the ground truth that are clustered into different classes by all base clustering, and the value of the CA matrix element representing their relationship is also 0. The missing connection rate refers to the ratio of missing edges to the edges of the ground truth. The missing connection rates from three real datasets are shown on the right side.

Existing ensemble clustering algorithms interpret the intrinsic relationship patterns of samples by constructing CA matrix. The CA matrix with symmetric properties describes the probability that samples belong to the same class. Therefore, consensus results can be obtained by using spectral clustering or hierarchical clustering on the CA matrix, but this leads to extremely low accuracy. Many ensemble clustering algorithms introduce various theories to improve the CA matrix and then improve the clustering performance. Some of these methods improve the quality of the CA matrix by applying weights to the CA matrix. For example, (Berikov and Pestunov 2017) applied weights determined by the evaluation function, (Huang, Wang, and Lai 2017) applied the weight calculated by the entropy value, and (Xu and Ding 2021) applied the weight with dual-granularity. Another approach involves designing a learning framework to enhance the quality of the CA matrix. (Jia et al. 2021) utilized low-rank tensor approximation to capture a refined CA matrix. (Zhou et al. 2023) proposed a method to enhance the CA matrix by active learning and self-paced learning. These methods focus on non-zero element value information in the

*Co-corresponding authors.

CA matrix, i.e., samples with edge connections. However, even if the element value of the CA matrix reaches the maximum or minimum (1 or 0, where all base clustering clusters two samples into the same class or different classes), it could still be misleading compared to the ground truth. This is due to the limitations of the base clustering algorithm, and it leads to the case of missing connections. Figure 1 shows the implications of missing connections. And Example 1 illustrates the challenges of missing connections.

Example 1. *Compared with the ground truth, the proportion of missing connections in the base clustering can reach up to 50% in real datasets. Existing ensemble models cannot classify missing connected samples into the same class, especially in the self-paced ensemble model (Zhou et al. 2020a), retaining the CA matrix element value of 0 is forced to be extended to the final CA matrix as a constraint.*

To tackle the above problem, this paper proposes a high-order topological model with adaptive weights for ensemble clustering (AWEC). The main support for the proposed method is: when two same class samples are clustered into different clusters by all base clustering, the value of the CA matrix element corresponding to their relationship is 0. This shows that in the view of the base clustering, there is no connection relationship between them. From the perspective of high-order connections, these two samples lack a first-order connection. However, they may establish a more intricate relationship (second-order connection) when considering a neighbor network. Hence, the introduction of high-order connection theory helps to compensate for missing connections, and combining multi-order connection information to learn an optimal connection matrix has the potential to improve the CA matrix. Furthermore, topology learning is embedded into the learning framework to further enhance the CA matrix. Specifically, topological similarity means that if two samples are similar, the similarity between the remaining samples and these two samples should be consistent.

Finally, this learning framework obtains a topology Z, and the ultimate consensus result is derived by applying spectral clustering (AWEC-S) or hierarchical clustering algorithm (AWEC-H). The main contributions of this paper are summarized as follows:

- This paper introduces high-order connectivity theory, enabling the learning of optimal connectivity matrices in ensemble clustering.
- Furthermore, building upon the learned optimal connection matrix, this paper utilizes topological similarity theory to explore the topological structure. Notably, this is the pioneering use of topological similarity learning in ensemble clustering, to the best of our knowledge.
- The comparison of experimental results with 9 state-of-the-art methods shows the superiority of the proposed method. Finally, we discuss parameter setting and optimal order selection.

Related Work

Ensemble clustering aims to derive a strong and robust consensus result by combining multiple weak base clustering results. The CA matrix is widely used in various ensemble clustering methods as a data object representing the relationship between samples. The CA matrix is proposed by (Fred and Jain 2005), and its definition is as follows:

$$\begin{aligned} A_{ij} &= \frac{1}{m}\sum_{k=1}^{m}\mathbb{I}\left(c_k\left(x_i\right), c_k\left(x_j\right)\right), \\ \mathbb{I}\left(c_k\left(x_i\right), c_k\left(x_j\right)\right) &= \begin{cases} 1, c_k\left(x_i\right) = c_k\left(x_j\right) \\ 0, c_k\left(x_i\right) \neq c_k\left(x_j\right) \end{cases} \end{aligned} \tag{1}$$

where A_{ij} represents the element value of the CA matrix, $c_k(x_i)$ represents the cluster of the instance x_i in the k-th base clustering result and m is the total number of base clustering results. The element values of the CA matrix can represent the probability that samples belong to the same class.

In general, the first-order connectivity of the graph explains the direct relationship between vertices. The second-order connectivity of the graph represents the neighbor network relationship between vertices. The more similar the neighborhood network of two vertices, the higher their second-order similarity. On this basis, it is extended to high-order connections, and its mathematical formula is as follows (Zhou et al. 2020b):

$$G_{ij}^{(2)} = \left(G_i\right)^{\top} G_j,\ \forall i, j \in [n]. \quad G^{(o)} = G^{(o-1)}G. \tag{2}$$

where $G^{(2)} \in \mathbb{R}^{n\times n}$ is the second-order connectivity matrix, G_i is the i-th row of the first-order connectivity matrix G, n represents the number of samples, and $G^{(o)}$ is o-order connectivity matrix.

High-dimensional data in the real world are often determined by low-dimensional structures (Chen, Li, and You 2020), generally linear or nonlinear structures. Recent methods for mining nonlinear structures are based on topology. Topological relationships can be propagated through highly reliable neighbors (Huang et al. 2022a,b). Topology learning is formulated as follows:

$$\begin{aligned} \min_{\mathbf{Z}^{(v)},\mathbf{S}} & \frac{1}{2}\sum_{v=1}^{m}\sum_{i,j,k=1}^{n}\mathbf{Q}_{jk}^{(v)}\left(\frac{\mathbf{Z}_{ij}^{(v)}}{\sqrt{\mathbf{D}_{jj}^{(v)}}} - \frac{\mathbf{Z}_{ik}^{(v)}}{\sqrt{\mathbf{D}_{kk}^{(v)}}}\right)^2 + \\ & \beta\left\|\mathbf{S} - \mu^{(v)}\mathbf{Z}^{(v)}\right\|_F^2 + \alpha\|\mathbf{Z}^{(v)} - \mathbf{I}\|_F^2 \\ \text{s.t. } & \mathbf{Z}_{ij}^{(v)} \geq 0,\ \left(\mathbf{Z}_i^{(v)}\right)^{\top}\mathbf{1} = 1,\ \mu^{(v)} \geq 0,\ \sum_{v=1}^{m}\mu^{(v)} = 1, \\ & \left(\mathbf{S}_i^{(v)}\right)^{\top}\mathbf{1} = 1,\ \mathbf{S}_{ij}^{(v)} \geq 0,\ \operatorname{rank}\left(\mathbf{L}_{\mathbf{S}}\right) = n - c, \end{aligned} \tag{3}$$

where $\mathbf{Q} \in \mathbb{R}^{n\times n}$ represents the similarity graph, $\mathbf{S}$ is the consensus graph, $\mathbf{Z}^{(v)}$ denotes the topological relationship matrix, and the constraint on $\mathbf{Z}^{(v)}$ guarantees that the sum of each column of $\mathbf{Z}^{(v)}$ is 1. $\mathbf{D}$ is the degree matrix of $\mathbf{Q}$, and $\mathbf{I}$ is the identity matrix. Both α and β are trade-off parameters. The rank constraint $\operatorname{rank}\left(\mathbf{L}_{\mathbf{S}}\right) = n - c$ on the Laplacian matrix $\mathbf{L}_{\mathbf{S}}$ ensures that $\mathbf{S}$ contains c connected components.

Our Proposed Methodology

First of all, we need to characterize high-order connection information of CA matrix. We calculate the neighborhood first-order connectivity matrix by the following formula:

$$G_{ij}^{(1)} = \begin{cases} k(\boldsymbol{x}_i,\ \boldsymbol{x}_j), & \text{if } \boldsymbol{x}_i \text{ and } \boldsymbol{x}_j \text{ are linked;} \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

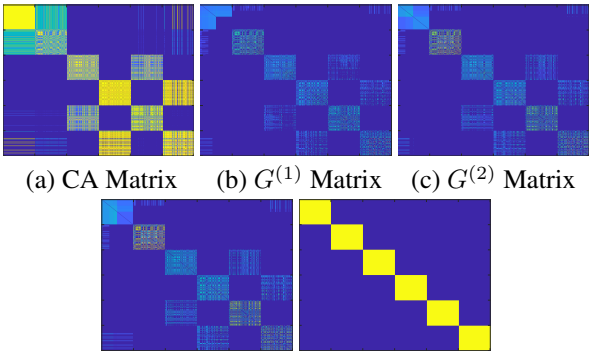
where $k(\cdot,\cdot)$ is a kernel function to measure the similarity between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ (Liang et al. 2020). In contrast to other graph learning methods, where $\boldsymbol{x}_i$ denotes the i-th sample, the proposed method uses the i-th row of the CA matrix A_i to denote $\boldsymbol{x}_i$, excluding the data features. This ensures that the proposed method maintains the definition of ensemble clustering. Next, we build high-order connection matrix according to Eq. (2). After obtaining the high-order connection matrix, a natural idea is to combine the connection information of each order and then minimize the learning to an optimal connection matrix $C \in \mathbb{R}^{n\times n}$, which is formulated as: $\min_C \sum_{i=1}^{o} \left\|C - G^{(i)}\right\|_F^2$, where o is the maximum order of the high-order connection matrix. However, it treats each order's connection information equally, ignoring its uniqueness.

A well-crafted mathematical definition should assign greater weights to good connection matrices while less weights to bad ones. Therefore, we adopt adaptive weights to achieve it. Next, we introduce the CA matrix enhancement assumption $A = C + E$ in the ECCMS model (Jia et al. 2023), which considers that the CA matrix in real data is the sum of pure CA matrix and noise matrix. We embed it into the high-order model with the following formula:

$$\begin{aligned} &\min_{C,E} \sum_{i=1}^{o} \left\|C - \alpha_i G^{(i)}\right\|_F^2 + \frac{\lambda}{2}\|E\|_F^2 \\ &\text{s.t. } A = C + E,\ 1 \geq C_{ij} \geq 0,\ C = C^\top, \\ &\qquad \sum_{i=1}^{o} \alpha_i = 1,\ \alpha_i \geq 0, \end{aligned} \tag{5}$$

where $E \in \mathbb{R}^{n\times n}$ is the noise matrix, and α_i denotes the weight of G^i. A is the original CA matrix, and the constraints on C preserve the symmetric properties and the range of matrix element values. After obtaining the optimal connection representation that absorbs the multi-order connection information, we further embed it as a topologically learned similarity graph to get the final objective function:

$$\begin{aligned} &\min_{C,E,Z} \sum_{i=1}^{o} \left\|C - \alpha_i G^{(i)}\right\|_F^2 + \frac{\lambda}{2}\|E\|_F^2 + \\ &\frac{1}{2}\sum_{i,j,k=1}^{n} C\left(\frac{Z_{ij}}{\sqrt{D_{jj}}} - \frac{Z_{ik}}{\sqrt{D_{kk}}}\right)^2 + \gamma\|Z - I\|_F^2 \\ &\text{s.t. } A = C + E,\ Z_i^\top \mathbf{1} = 1,\ Z_{ij} \geq 0,\ C = C^\top \\ &\text{rank}\left(\mathbf{L_z}\right) = n - c,\ 1 \geq C_{ij} \geq 0,\ \sum_{i=1}^{o} \alpha_i = 1,\ \alpha_i \geq 0. \end{aligned} \tag{6}$$



(a) CA Matrix (b) $G^{(1)}$ Matrix (c) $G^{(2)}$ Matrix
(d) $G^{(3)}$ Matrix (e) Ground Truth

Figure 2: CA matrix and multi-order connectivity matrix visualization.

where $D \in \mathbb{R}^{n\times n}$ is the degree matrix of topology $Z \in \mathbb{R}^{n\times n}$, and the last term $\gamma\|Z - I\|_F^2$ is to avoid trivial solutions. The rank constraint on the Laplacian matrix L_z ensures that Z exactly contains c connected components.

Based on Eq. (2) and Eq. (4), we compute the CA matrix and multi-order connection matrices in **Control** dataset, as visualized in Figure 2. Notably, the initial CA matrix exhibits more noise compared to the ground truth, while the other connectivity matrix is cleaner.

Optimization

We apply the Alternating Direction Method of Multipliers (ADMM) method (Boyd et al. 2011) to solve the relevant variables in the objective function. We introduce an auxiliary variable J for optimization. We can get the following augmented Lagrangian function:

$$\begin{aligned} \mathcal{L} = &\sum_{i=1}^{o} \left\|C - \alpha_i G^{(i)}\right\|_F^2 + \frac{\lambda}{2}\|E\|_F^2 + \gamma\|Z - I\|_F^2 + \\ &\frac{1}{2}\sum_{i,j,k=1}^{n} C_{j,k}\left(\frac{Z_{ij}}{\sqrt{D_{jj}}} - \frac{Z_{ik}}{\sqrt{D_{kk}}}\right)^2 + \langle Y_1, A - C - E\rangle \\ &+ \langle Y_2, C - J\rangle + \frac{\mu}{2}\left(\|A - C - E\|_F^2 + \|C - J\|_F^2\right). \end{aligned} \tag{7}$$

where Y_1 and Y_2 are Lagrangian multipliers. $\langle\cdot,\cdot\rangle$ denotes the inner product. Next, we need to fix some variables and then update the target variable.

Updating C To update variable C, we need to fix all other variables and then re-write the formula as follows:

$$\begin{aligned} C = &\text{argmin}_C \sum_{i=1}^{o} \left\|C - \alpha_i G^{(i)}\right\|_F^2 + \langle Y_2, C - J\rangle \\ &+ \frac{1}{2}\sum_{i,j,k=1}^{n} C_{j,k}\left(\frac{Z_{ij}}{\sqrt{D_{jj}}} - \frac{Z_{ik}}{\sqrt{D_{kk}}}\right)^2 + \frac{\mu}{2}\|C - J\|_F^2 \\ &+ \frac{\mu}{2}\|A - C - E\|_F^2 + \langle Y_1, A - C - E\rangle. \end{aligned} \tag{8}$$

For Eq. (8), it is an unconstrained problem, so it can be derived directly to get the following results:

$$C = (2oI + 2\mu I)^{-1} \left(\mu \left(A - E + \frac{Y_1}{\mu} \right) + \mu \left(J - \frac{Y_2}{\mu} \right) + FF^T + 2 \sum_{i=1}^{o} \alpha_i G^{(i)} \right), \tag{9}$$

where $F = D^{-\frac{1}{2}} Z$.

Updating J Similarly, we fix other variables and update variable J, the sub-problem is as follows:

$$J = \text{argmin}_J \langle Y_2, C - J \rangle + \frac{\mu}{2} \|C - J\|_F^2 \quad \text{s.t. } 1 \geq J \geq 0,\ J = J^\top. \tag{10}$$

For Eq. (10), according to (Jia et al. 2023), its solution has an element-wise truncation result:

$$J = \min \left(\max \left(\frac{\left(C + \frac{Y_2}{\mu} \right)}{2} + \frac{\left(C^T + \frac{Y_2^\top}{\mu} \right)}{2}, 0 \right), 1 \right). \tag{11}$$

Updating E We remove other variables and retain items related to variable E, and we can get the sub-problem of E as follows:

$$E = \text{argmin}_E \frac{\lambda}{2} \|E\|_F^2 + \frac{\mu}{2} \|A - C - E\|_F^2 + \langle Y_1, A - C - E \rangle. \tag{12}$$

Regarding Eq. (12), we directly derive it and make the derivative 0 to get the result as follows:

$$E = \frac{\mu (A - C) + Y_1}{\lambda + \mu}. \tag{13}$$

Updating Z The sub-problem of Z are as follows:

$$Z = \text{argmin}_Z \frac{1}{2} \sum_{i,j,k=1}^{n} C_{j,k} \left(\frac{Z_{ij}}{\sqrt{D_{jj}}} - \frac{Z_{ik}}{\sqrt{D_{kk}}} \right)^2 + \gamma \|Z - I\|_F^2 \quad \text{s.t. } Z_i^\top \mathbf{1} = 1,\ Z_{ij} \geq 0,\ \text{rank}\left(\mathbf{L_z}\right) = n - c. \tag{14}$$

Solving the sub-problem of Z is a bit more complicated because it involves rank constraints. According to Ky Fan's Theorem (Fan 1949), rank constraints $\text{rank}\left(L_z\right) = n - c$ can be transformed into the following form:

$$\min Tr\left(H^\top L_Z H\right) \quad \text{s.t.} \quad H \in \mathbb{R}^{n \times c},\ H^\top H = I. \tag{15}$$

Next, we can re-write Eq. (14) as follows:

$$Z = \text{argmin}_Z \frac{1}{2} \sum_{i,j,k=1}^{n} C_{j,k} \left(\frac{Z_{ij}}{\sqrt{D_{jj}}} - \frac{Z_{ik}}{\sqrt{D_{kk}}} \right)^2 + \gamma \|Z - I\|_F^2 + 2\delta \,\text{Tr}\left(H^\top L_Z H\right) \quad \text{s.t. } Z_i^\top \mathbf{1} = 1,\ Z_{ij} \geq 0, \tag{16}$$

where δ is a self-tuned parameter. We first optimize the variable H. Since Eq. (15) is essentially a spectral problem, we only need to calculate the c eigenvectors of L_Z corresponding to the c smallest eigenvalues. For the optimization of the variable Z in Eq. (15), we adopt the solution process in (Huang et al. 2022b), and the detailed solution process is shown in Algorithm 1.

Updating α_i With other variables fixed, the sub-problem for variable α_i is as follows:

$$\alpha_i = \text{argmin}_{\alpha_i} \sum_{i=1}^{o} \left\| C - \alpha_i G^{(i)} \right\|_F^2 \quad \text{s.t. } \sum_{i=1}^{o} \alpha_i = 1,\ \alpha_i \geq 0. \tag{17}$$

We obtain the following Lagrangian function about Eq. (17):

$$\mathcal{L}(\alpha_i) = \left\| C - \alpha_i G^{(i)} \right\|_F^2 + \theta \left(\sum_{i=1}^{o} \alpha_i - 1 \right) \tag{18}$$

where θ means the Lagrange multiplier. Directly, we take the derivative of Eq. (18) and make it equal to 0, then we can get the following result:

$$\alpha_i = \frac{2\mathbf{Tr}\left(C G^{(i)}\right) - \theta}{2\mathbf{Tr}\left(G^{(i)} \left(G^{(i)}\right)^\top\right)} \tag{19}$$

Updating Y_1**,** Y_2**, and** μ Lagrange multipliers Y_1, Y_2, and μ in ADMM algorithm are updated by:

$$\begin{aligned} Y_1 &= Y_1 + \mu(A - C - E), \\ Y_2 &= Y_2 + \mu(C - J), \\ \mu &= min(\rho_1 \mu,\ \mu_{max}). \end{aligned} \tag{20}$$

where $\rho_1 > 1$ is to ensure the number of convergence is lower and μ_{max} is the maximum value of μ (Chen, Xiao, and Zhou 2019).

The overall flow of our proposed AWEC is shown in Algorithm 2. In short, the most significant difference between the proposed method and other ensemble methods is that an optimal connection is learned from multi-order connection information in an adaptive weight manner, and the topology Z is learned via denoising optimal connection matrix.

Complexity Analysis

In total, there are five variables requiring updates. First, for the update of variable C, because it involves matrix inverse operation and matrix multiplication, its time complexity is $O\left(n^3\right)$. However, since the inverse operation is a fixed item, it can be calculated in advance to improve certain time efficiency. Regarding the variables E, J, Y_1, and Y_2, the time complexity is $O\left(n^2\right)$. The time complexity of variable α_i is $O\left(n^3\right)$. To update variable Z, variable H needs to be updated first, so the total time complexity of these two parts is $O\left(n^2\right)$. Overall, the time complexity of iteratively updating variables is $O\left(n^3\right)$.

Algorithm 1: ALM update Z

Input: a nonzero vector q , $G' = \left(I - D^{-\frac{1}{2}} C D^{-\frac{1}{2}}\right) + \gamma I$, $b = 2\gamma I_i - \delta \left\| h_i - h_\cdot \right\|_2^2$, $\rho_2 = 1.5$, $\eta = 10$
Output: Topology Z

1: Eq. (16) can be rewritten as $\min_{z_{ij} \geq 0, z_i^\top 1 = 1} z_i^\top G' z_i - z_i^\top b$.
2: Obtain the augmented Lagrangian function.
3: **while** not converged **do**
4: Update p by $p = z_i - \frac{1}{\eta}((G')^\top z_i + q)$.
5: Update z_i by $\min_{z_{ij} \geq 0, z_i^\top \mathbf{1} = 1} \left\| z_i - p + \frac{1}{\eta} q + \frac{G' p - b}{\eta} \right\|_2^2$, which can be solved according to (Huang, Nie, and Huang 2015).
6: Update η by $\eta = \rho_2 \eta$.
7: Update q by $q = q + \eta(z_i - p)$.
8: **end while**
9: Obtain the topology Z

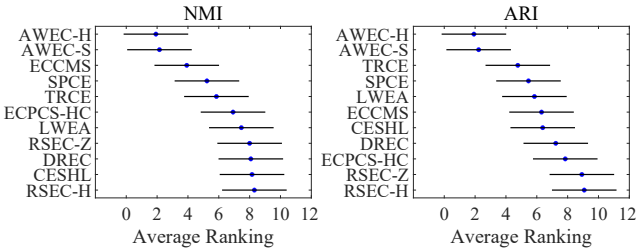
Algorithm 2: AWEC

Input: CA matrix A, Noise penalty parameter λ, γ
Initialization: $C = J = Z = I$, $E = 0$, Y_1, Y_2, $\mu = 0.1$, $\mu_{max} = 10^8$, $\rho_1 = 1.2$, and $\epsilon = 10^{-2}$.
Output: Consensus result S.

1: **while** not converged **do**
2: Update C by Eq. (9).
3: Update J by Eq. (11).
4: Update E by Eq. (13).
5: Update Z by Algorithm 1.
6: Update α by Eq. (19).
7: Update the multipliers by Eq. (20).
8: Check the convergence conditions:$||A - C - E||_\infty < \epsilon$ and $||C - J||_\infty < \epsilon$.
9: **end while**
10: Apply the average-link hierarchical agglomerative clustering method (AWEC-H) and spectral clustering (AWEC-S) to Z.
11: Obtain consensus clustering result S.

Experiments

We conduct extensive experiments on 14 real datasets from different domains. Characteristics of these datasets are provided in Table 1. The samples of the dataset range from 100 to 11000, and the number of clusters ranges from 2 to 36. We randomly run the k-means algorithm 100 times on each dataset (http://archive.ics.uci.edu/datasets) (Huang, Wang, and Lai 2017; Zhou, Zheng, and Pan 2019; Yu et al. 2022) to generate the base clustering result set. We set the ensemble size $M = 20$, conduct 10 repeated experiments with different base clustering combinations, and finally report the average value. All compared methods use the same combination of base clustering. For the number of neighbors parameter in Eq. (4), we set it to $0.5s$ in all datasets, where $s = n/c$ represents the average sample number in each category (Zhou et al. 2020b). We use 9 state-of-the-art ensemble clustering approaches, including DREC (Zhou, Zheng, and Pan 2019), LWEA (Huang, Wang, and Lai 2017), CESHL (Zhou et al. 2022), SPCE (Zhou et al. 2020a), TRCE (Zhou et al. 2021), ECCMS (Jia et al. 2023), ECPCS-HC (Huang et al. 2021), RSEC-Z and RSEC-H (Tao et al. 2019), for comparison. KM and KM-best is the average and the optimal result of the base clustering set generated by k-means. We use three popular metrics to evaluate clustering performance: Normalized Mutual Information (NMI), Adjusted Rand Index (ARI), and F1-score.



(a) NMI (b) ARI

Figure 3: Friedman Test on two metrics. The blue dots indicate the average rank, and the overlapping area of the lines indicates whether the methods differ significantly.

Experimental Results

Tables 2, 3, and 4 show the experimental results of all algorithms on three metrics. It is clear that the proposed method outperforms all compared methods in most datasets. Specifically, the proposed method consistently outperforms the compared methods across all datasets, as evaluated by the ARI metric. The proposed method achieves the best results in the NMI metric for most datasets, except for Coil20 and Control. Regarding F1-score, the proposed method also performs the best, except for Vertebral2c. Overall, the performance of the proposed method is extremely superior. Furthermore, the results of AWEC-H and AWEC-S algorithms yield comparable outcomes across most datasets, with an average error of less than 2% across all datasets. This demonstrates that while AWEC-H exhibits superiority, it is not consistently optimal. The minor error indicates that the partitioning method has limited impact on the ultimate consensus result. And the closest similarity to the proposed method is the ECCMS method, while ECCMS exceeds the proposed method only with the NMI metric on Control. In other datasets, the proposed method surpasses ECCMS, especially in Vertebral3c. The ARI metric of the proposed method is improved by 38% compared with ECCMS. Finally, in large-scale dataset such as USPS, our method is the best in all three metrics, which shows that the proposed method is applicable to large-scale datasets. The superiority can be confirmed by the Friedman test in Figure 3.

Parameter Study

AWEC has two main parameters: the noise regularization parameter λ and the parameter γ. We perform a 6*6 grid search for λ in the set $\{0.01, 0.02, 0.04, 0.08, 0.1, 0.2\}$ and

No.	Dataset	Instance	Feature	Class	No.	Dataset	Instance	Feature	Class
D1	Appendicitis	106	7	2	D8	IS	2310	19	7
D2	Breast Cancer	683	9	2	D9	Led7digit	500	7	10
D3	Binalpha	1404	320	36	D10	Ionosphere	351	34	2
D4	Breast Tissue	106	9	6	D11	MF	2000	649	10
D5	Coil20	1440	1024	20	D12	Vertebral3c	310	6	3
D6	Control	600	60	6	D13	Vertebral2c	310	6	2
D7	Fertility	100	9	2	D14	USPS	11000	256	10

Table 1: Characteristics of datasets.

Dataset	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14
KM	0.1502	0.4352	0.2681	0.1582	0.5854	0.5404	0.0115	0.3440	0.4065	0.1845	0.3743	0.2164	0.0638	0.2155
KM-best	0.3787	0.8520	0.3032	0.2051	0.6779	0.6174	0.0633	0.5670	0.4934	0.3363	0.5045	0.3246	0.1113	0.3388
CESHL	0.3802	0.8704	0.2843	0.1167	0.6365	0.5554	0.0847	0.5251	0.4540	0.1787	0.5102	0.0834	0.1576	0.4565
ECCMS	0.3093	0.8236	0.3042	0.1143	0.6163	0.6060	0.0758	0.4783	0.4420	0.1918	0.5290	0.2229	0.2405	0.5263
SPCE	0.4177	0.8460	0.2840	0.1868	0.6752	0.6068	0.1008	0.4693	0.3880	0.2482	0.5235	0.2847	0.0669	0.4252
DREC	0.1242	0.8279	0.3049	0.1005	0.5320	0.5584	0.0688	0.5244	0.4354	0.1737	0.4939	0.3377	0.0228	0.5276
TRCE	0.4533	0.8888	0.2968	0.1730	0.6618	0.5911	0.0848	0.5202	0.4781	0.1834	0.5268	0.0934	0.0101	0.4703
LWEA	0.2770	0.8221	0.3018	0.1102	0.6268	0.5776	0.0984	0.5222	0.4371	0.2065	0.5401	0.2303	0.2076	0.5373
ECPCS-HC	0.2991	0.8644	0.2710	0.1239	0.5582	0.5524	0.1069	0.5192	0.3998	0.1611	0.4811	0.0180	N/A	0.4442
RSEC-Z	0.3910	0.5176	0.2179	0.1266	0.5366	0.5328	0.0703	0.3588	0.3652	0.1082	0.5105	0.2369	0.2231	0.4261
RSEC-H	0.4030	0.6143	0.2358	0.0900	0.4730	0.5543	0.0703	0.3757	0.3559	0.1247	0.5105	0.2710	0.1677	0.4339
AWEC-S	0.3723	0.8888	0.3148	0.2147	0.6759	**0.6356**	0.1025	**0.5475**	**0.4896**	0.2040	**0.5844**	**0.6055**	**0.2706**	**0.5607**
AWEC-H	**0.4659**	**0.8894**	**0.3191**	**0.2794**	**0.6919**	0.6326	**0.1202**	0.4991	0.4773	**0.3635**	0.5723	0.5802	0.2700	0.5584

Table 2: Clustering performance measured by ARI.

Dataset	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14
KM	0.1707	0.4705	0.5746	0.3546	0.7853	0.7075	0.0405	0.6001	0.5469	0.2120	0.5989	0.3862	0.1899	0.5346
KM-best	0.2415	0.7546	0.6019	0.4364	0.8208	0.7723	0.0769	0.6712	0.6154	0.3255	0.6435	0.4353	0.2553	0.5785
CESHL	0.2144	0.7783	0.5688	0.2530	0.8011	0.6827	0.0275	0.6226	0.5496	0.1337	0.6442	0.2060	0.2056	0.6085
ECCMS	0.1959	0.7230	0.6000	0.3066	0.8430	**0.7762**	0.0407	0.6499	0.5577	0.1599	0.6938	0.3893	0.2826	0.6643
SPCE	0.2303	0.7444	0.6085	0.3800	**0.8546**	0.7452	0.0579	0.5873	0.5040	0.2002	0.6609	0.3831	0.1617	0.5675
DREC	0.1091	0.7197	0.5944	0.2432	0.7432	0.6868	0.0257	0.6341	0.5400	0.1278	0.6377	0.4778	0.1463	0.6388
TRCE	0.2561	0.8124	0.5946	0.3010	0.8272	0.7284	0.0288	0.6146	0.5746	0.1389	0.6575	0.2264	0.1546	0.6162
LWEA	0.1569	0.7136	0.5831	0.2487	0.7917	0.7068	0.0283	0.6246	0.5318	0.1524	0.6661	0.3710	0.2529	0.6512
ECPCS-HC	0.1819	0.7819	0.5846	0.3098	0.7892	0.7359	0.0393	0.6428	0.5414	0.1248	0.6373	0.2207	N/A	0.6169
RSEC-Z	0.2345	0.5496	0.5585	0.2743	0.8078	0.7192	0.0359	0.5764	0.5092	0.1060	0.6706	0.3050	0.2658	0.6158
RSEC-H	0.2439	0.6230	0.5657	0.2316	0.7956	0.7179	0.0387	0.6006	0.4984	0.1150	0.6638	0.3214	0.2083	0.6153
AWEC-S	0.2182	**0.8225**	0.6074	0.3761	0.8506	0.7560	0.0824	**0.6687**	**0.5866**	0.1595	**0.7011**	**0.5278**	0.3112	**0.6865**
AWEC-H	**0.2989**	0.8173	**0.6105**	**0.4344**	0.8416	0.7538	**0.1021**	0.6459	0.5795	**0.3036**	0.6972	0.5054	**0.3137**	0.6775

Table 3: Clustering performance measured by NMI.

γ in the set $\{0.1, 0.5, 1, 5, 10, 50\}$ to learn the optimal parameter combination. Figure 4 shows the NMI results of the proposed method in different parameter combinations. In general, the optimal results of the proposed method are affected by the parameters to a certain extent. Still, we can find some rules by observing the parameter sensitivity maps and reducing the parameter search range of the optimal result. Specifically, we observe that the subset of γ corresponding to optimal results is $\{5, 10, 50\}$. Taking both parameters into account, an appropriate combination is γ from $\{5, 10, 50\}$ and λ from $\{0.08, 0.1, 0.2\}$.

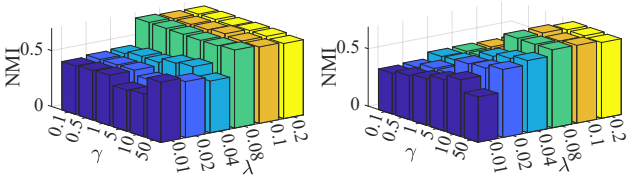
Table 5 shows the effect of maximum order o on ensemble performance. We find that ensemble performance is not increased as the maximum order becomes larger. Whether AWEC-H or AWEC-S, the optimal performance corresponds to the maximum order is no more than 3. To

Dataset	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14
KM	0.4815	0.6021	0.2891	0.3392	0.6059	0.6264	0.3353	0.3996	0.4651	0.4823	0.4229	0.4006	0.3420	0.2495
KM-best	0.7804	0.9331	0.3228	0.3636	0.6922	0.6881	0.6635	0.6150	0.5449	0.6328	0.5480	0.6153	0.7166	0.3995
CESHL	0.7905	0.9408	0.3082	0.3218	0.6567	0.6447	0.7083	0.6015	0.5156	0.6052	0.5642	0.5207	0.6918	0.5196
ECCMS	0.7900	0.9209	0.3272	0.3205	0.6393	0.6826	0.6783	0.5702	0.5067	0.6999	0.5834	0.5592	0.7060	0.5793
SPCE	0.8300	0.9304	0.3000	0.3623	0.6894	0.6771	0.7653	0.5563	0.4278	0.6167	0.5762	0.5380	0.7166	0.4893
DREC	0.6938	0.9227	0.3255	0.3108	0.5605	0.6416	0.6682	0.5975	0.4975	0.6034	0.5487	0.5938	0.6292	0.5773
TRCE	0.8484	0.9486	0.3175	0.3582	0.6808	0.6642	0.8806	0.5971	0.5321	0.7000	0.5787	0.5608	**0.7192**	0.5332
LWEA	0.7336	0.9201	0.3239	0.3178	0.6476	0.6557	0.7132	0.6047	0.5022	0.6218	0.5905	0.5781	0.6727	0.5886
ECPCS-HC	0.7433	0.9382	0.2977	0.3260	0.5849	0.6462	0.7173	0.5975	0.4727	0.5964	0.5386	0.4914	N/A	0.5121
RSEC-Z	0.8234	0.8042	0.2447	0.3237	0.5647	0.6249	0.8203	0.4792	0.4441	0.6808	0.5649	0.6004	0.6356	0.4947
RSEC-H	0.8208	0.8409	0.2618	0.3116	0.5075	0.6442	0.8179	0.4944	0.4396	0.6773	0.5657	0.6162	0.6480	0.5009
AWEC-S	0.8021	0.9486	0.3347	0.3782	0.6930	**0.6977**	0.8694	**0.6173**	**0.5417**	0.6663	**0.6270**	**0.7525**	0.7040	**0.6095**
AWEC-H	**0.8511**	**0.9490**	**0.3385**	**0.4253**	**0.7077**	0.6951	**0.8881**	0.5827	0.5305	**0.7310**	0.6164	0.7383	0.7166	0.6059

Table 4: Clustering performance measured by F1-score.

	Max. Order	1st	2nd	3rd	4th	5th
D1	AWEC-S	0.2035	**0.2182**	0.1926	0.1901	0.1986
	AWEC-H	0.2510	**0.2989**	0.1824	0.2164	0.1987
D12	AWEC-S	0.4933	**0.5278**	0.5145	0.5004	0.5015
	AWEC-H	0.4854	0.5054	**0.5118**	0.5112	0.5075
D13	AWEC-S	0.3097	**0.3112**	0.3082	0.3104	0.3089
	AWEC-H	0.2988	0.3137	**0.3141**	0.3075	0.3067

Table 5: NMI w.r.t. maximum order o.



(a) MF (b) USPS

Figure 4: NMI w.r.t. γ and λ.

Metric	Method	D1	D5	D11
NMI	AWEC-H	**0.2989**	**0.8416**	**0.6972**
	w/o Topo	0.1640	0.8183	0.6547
	w/o High-order	0.1524	0.8146	0.6007

Table 6: NMI w.r.t. Ablation results.

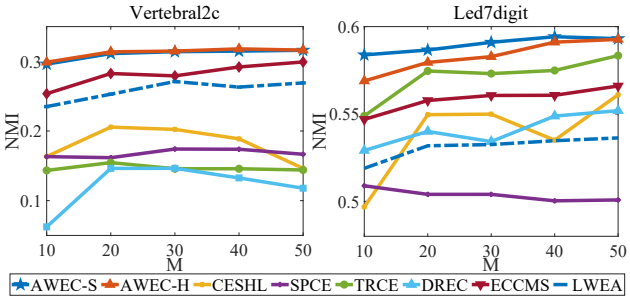


Figure 5: Algorithm performance comparison on different ensemble sizes M.

improve time efficiency, the better choice for the maximum order of the proposed approach is 2.

Ablation Study

Table 6 shows the ablation experimental results of AWEC. We observe that our proposed AWEC method, when excluding the topological module and high-order learning module, shows a significant performance degradation. This once again demonstrates the effectiveness of AWEC in introducing topology and high-order connections.

The Influence of Ensemble Size

Figure 5 shows the effect of five ensemble sizes, $\mathbf{M} = \{10, 20, 30, 40, 50\}$, with different algorithms on Vertebra2c and Led7digit. In general, ensemble performance is positively correlated with ensemble size. The proposed method basically maintains this property. At the same time, it is clear that our proposed AWEC outperforms the compared methods across all ensemble sizes.

Conclusion

In this paper, we propose a topological ensemble model based on high-order connections with adaptive weights. The proposed method learns the optimal connection matrix from multi-order connection matrices with adaptive weights, uses the optimal connection as the input for topology learning, and finally integrates it into a dynamic and unified learning framework. Extensive experimental results show that the proposed method surpasses many state-of-the-art methods. However, the optimal results of the proposed method in some datasets are affected by parameters. Therefore, improving the robustness of the algorithm is what we will explore in the future.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (61972268) and the Ministry of Education of Humanities and Social Science Project (19YJAZH047).

References

Berikov, V.; and Pestunov, I. 2017. Ensemble Clustering Based on Weighted Co-association Matrices: Error Bound and Convergence Properties. *Pattern Recognition*, 63: 427–436.

Boyd, S.; Parikh, N.; Chu, E.; Peleato, B.; Eckstein, J.; et al. 2011. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundations and Trends® in Machine learning*, 3(1): 1–122.

Chen, L.; Wang, K.; Li, M.; Wu, M.; Pedrycz, W.; and Hirota, K. 2022. K-means Clustering-based Kernel Canonical Correlation Analysis for Multimodal Emotion Recognition in Human–robot Interaction. *IEEE Transactions on Industrial Electronics*, 70(1): 1016–1024.

Chen, Y.; Li, C.-G.; and You, C. 2020. Stochastic Sparse Subspace Clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4155–4164.

Chen, Y.; Xiao, X.; and Zhou, Y. 2019. Low-rank Quaternion Approximation for Color Image Processing. *IEEE Transactions on Image Processing*, 29: 1426–1439.

Dai, C.; Wu, J.; Pi, D.; Cui, L.; Johnson, B.; and Becker, S. I. 2022. Electroencephalogram Signal Clustering With Convex Cooperative Games. *IEEE Transactions on Knowledge and Data Engineering*, 34(12): 5755–5769.

Elhamifar, E.; and Vidal, R. 2013. Sparse Subspace Clustering: Algorithm, Theory, and Applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11): 2765–2781.

Fan, K. 1949. On a theorem of Weyl concerning eigenvalues of linear transformations I. *Proceedings of the National Academy of Sciences*, 35(11): 652–655.

Fred, A. L.; and Jain, A. K. 2005. Combining Multiple Clusterings using Evidence Accumulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(6): 835–850.

Huang, D.; Wang, C.-D.; and Lai, J.-H. 2017. Locally Weighted Ensemble Clustering. *IEEE Transactions on Cybernetics*, 48(5): 1460–1473.

Huang, D.; Wang, C.-D.; Peng, H.; Lai, J.; and Kwoh, C.-K. 2021. Enhanced Ensemble Clustering via Fast Propagation of Cluster-wise Similarities. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 51(1): 508–520.

Huang, J.; Nie, F.; and Huang, H. 2015. A New Simplex Sparse Learning Model to Measure Data Similarity for Clustering. In *Proceedings of the 24th International Conference on Artificial Intelligence*, 3569–3575.

Huang, S.; Tsang, I.; Xu, Z.; Lv, J.; and Liu, Q.-H. 2022a. Multi-View Clustering on Topological Manifold. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 6944–6951.

Huang, S.; Wu, H.; Ren, Y.; Tsang, I.; Xu, Z.; Feng, W.; and Lv, J. 2022b. Multi-view Subspace Clustering on Topological Manifold. *Advances in Neural Information Processing Systems*, 35: 25883–25894.

Jia, Y.; Liu, H.; Hou, J.; and Zhang, Q. 2021. Clustering Ensemble meets Low-rank Tensor Approximation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 7970–7978.

Jia, Y.; Tao, S.; Wang, R.; and Wang, Y. 2023. Ensemble Clustering via Co-association Matrix Self-Enhancement. *IEEE Transactions on Neural Networks and Learning Systems*, 1–12.

Liang, W.; Zhou, S.; Xiong, J.; Liu, X.; Wang, S.; Zhu, E.; Cai, Z.; and Xu, X. 2020. Multi-view Spectral Clustering with High-order Optimal Neighborhood Laplacian Matrix. *IEEE Transactions on Knowledge and Data Engineering*, 34(7): 3418–3430.

Liu, A.-A.; Su, Y.-T.; Nie, W.-Z.; and Kankanhalli, M. 2017. Hierarchical Clustering Multi-Task Learning for Joint Human Action Grouping and Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(1): 102–114.

Liu, G.; Lin, Z.; Yan, S.; Sun, J.; Yu, Y.; and Ma, Y. 2012. Robust Recovery of Subspace Structures by Low-rank Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1): 171–184.

Rodriguez, A.; and Laio, A. 2014. Clustering by Fast Search and Find of Density Peaks. *Science*, 344(6191): 1492–1496.

Tao, Z.; Liu, H.; Li, S.; Ding, Z.; and Fu, Y. 2019. Robust Spectral Ensemble Clustering via Rank Minimization. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 13(1): 1–25.

Wang, Q.; Chen, M.; Nie, F.; and Li, X. 2020. Detecting Coherent Groups in Crowd Scenes by Multiview Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(1): 46–58.

Wang, Y.; Wu, L.; Lin, X.; and Gao, J. 2018. Multiview Spectral Clustering via Structured Low-rank Matrix Factorization. *IEEE Transactions on Neural Networks and Learning Systems*, 29(10): 4833–4843.

Xu, L.; and Ding, S. 2021. Dual-granularity Weighted Ensemble Clustering. *Knowledge-Based Systems*, 225: 107124.

Yu, Z.; Wang, D.; Meng, X.-B.; and Chen, C. L. P. 2022. Clustering Ensemble Based on Hybrid Multiview Clustering. *IEEE Transactions on Cybernetics*, 52(7): 6518–6530.

Zhou, J.; Zheng, H.; and Pan, L. 2019. Ensemble Clustering Based on Dense Representation. *Neurocomputing*, 357: 66–76.

Zhou, P.; Du, L.; Liu, X.; Shen, Y.-D.; Fan, M.; and Li, X. 2020a. Self-paced Clustering Ensemble. *IEEE Transactions on Neural Networks and Learning Systems*, 32(4): 1497–1511.

Zhou, P.; Du, L.; Shen, Y.-D.; and Li, X. 2021. Tri-level Robust Clustering Ensemble with Multiple Graph Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 11125–11133.

Zhou, P.; Sun, B.; Liu, X.; Du, L.; and Li, X. 2023. Active Clustering Ensemble with Self-Paced Learning. *IEEE Transactions on Neural Networks and Learning Systems*, 1–15.

Zhou, P.; Wang, X.; Du, L.; and Li, X. 2022. Clustering Ensemble via Structured Hypergraph Learning. *Information Fusion*, 78: 171–179.

Zhou, S.; Liu, X.; Liu, J.; Guo, X.; Zhao, Y.; Zhu, E.; Zhai, Y.; Yin, J.; and Gao, W. 2020b. Multi-view Spectral Clustering with Optimal Neighborhood Laplacian Matrix. In *Proceedings of the AAAI conference on Artificial Intelligence*, volume 34, 6965–6972.