

EvoFMVC: Trusted Federated Multi-View Clustering with Evolutionary Fusion

Li Zhang^{1*}, Pinhan Fu^{2*}, Li Lv¹, Qian Guo³, Liang Du¹, Xinyan Liang^{1†}

¹Institute of Big Data Science and Industry, Shanxi University

²School of Computer Science, Wuhan University

³Shanxi Key Laboratory of Big Data Analysis and Parallel Computing, School of Computer Science and Technology, Taiyuan University of Science and Technology

{z11370055, fupinhan168, lvli924, czguoqian, liangxinyan48}@163.com, duliang@sxu.edu.cn

Abstract

With the growing demand for decentralized collaborative analysis of privacy-sensitive data, federated multi-view clustering (FMVC) has attracted widespread attention due to its ability to balance privacy protection and collaborative modeling. However, current methods still face the following challenges: (1) Clients need to frequently upload high-dimensional data such as model parameters or graph structures, resulting in high communication costs; (2) The structured data uploaded often contains semantic features and has a high risk of being inverted; (3) The server usually merges the data from all clients with the fixed fusion rule, which may result in a suboptimized clustering result when there exist low-quality clients. To address the issues, we propose a new trusted federated multi-view clustering framework (EvoFMVC) that introduces three key innovations: First, lightweight trusted evidence serves as a compact communication medium, significantly reducing overhead compared to conventional model parameters or graph structures. Second, trusted evidences express clustering results in the form of probability distribution, which avoids the risk of structured information being easily inverted. Lastly, we formalize the server-side aggregation process as a neural architecture search (NAS) task where the server flexibly uses different fusion operators to filter and fuse necessary views through evolutionary algorithms, which significantly improves the fusion effect and model performance. Experimental results on multiple datasets show that our method is superior to existing FMVC methods in terms of clustering accuracy and communication efficiency.

Code — <https://github.com/xxx11ovo/EvoFMVC>

1 Introduction

With the continuous growth of the demand for data privacy protection, federated learning has gradually become an important way to analyze cross-institutional data due to its distributed collaboration mechanism and privacy protection characteristics (Huang et al. 2023). However, data in actual scenarios often have multiple different feature views. The federated learning method of a single view is difficult to

*These authors contributed equally.

†The corresponding author.

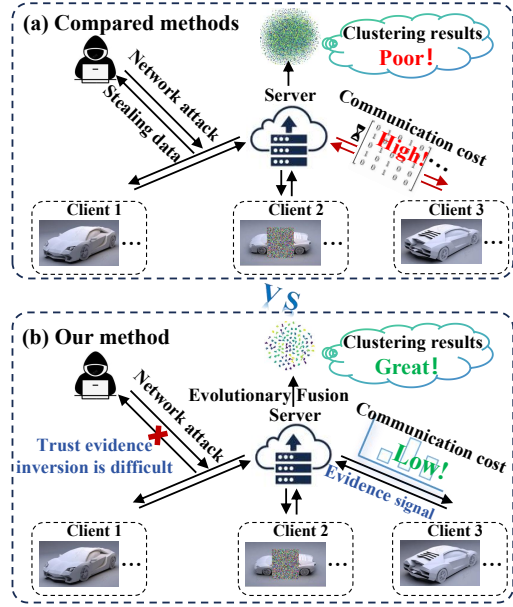


Figure 1: Comparison between existing methods and the proposed EvoFMVC.

fully capture the complexity of the data, and since different views are often stored in different institutional devices, communication costs also need to be considered (Fang and Ye 2025). Therefore, federated multi-view clustering was proposed to achieve multi-view collaborative analysis under privacy protection.

This paradigm, however, naturally brings three intertwined challenges: preserving data privacy, reducing communication cost, and effectively aggregating multi-view features. To tackle them, researchers have explored various solutions in recent years. In terms of privacy preservation, existing FMVC approaches often employ feature encoding, latent representation learning, or pseudo-label generation to avoid direct exposure of raw features. Such as FedDMC (Chen et al. 2023), which uses deep autoencoders to extract latent representations and constructs sample-level pseudo-labels to guide local updates. TensorFMVC (Feng et al. 2024) also protects privacy by transmitting only cluster

assignment matrices via a tensor factorization framework, rather than sharing raw features or embeddings. In terms of communication efficiency, many methods focus on reducing the size or frequency of data exchange, often via dimensionality reduction or model compression. For example, the improved FedDMC (Mu et al. 2024) leverages PCA-based compression and malicious client filtering to minimize communication overhead. Similarly, TensorFMVC (Feng et al. 2024) employs anchor-based affinity matrices and compact local representations to further reduce communication costs. In terms of server-side aggregation, once client updates are collected, various strategies are used to integrate information, such as uniform averaging or adaptive weighting. For example, Sun et al. (Sun et al. 2025) proposes a graph-based FMVC framework MGCD that extracts consistency and specificity representations at each client to construct view-specific diversity graphs, fused by simple averaging to produce a global consistency graph for spectral clustering.

Although the aforementioned methods have advanced FMVC along the axes of privacy preservation, communication efficiency, and information aggregation, core challenges in balancing these aspects persist in practical federated settings. (1) **High communication overhead.** Most FMVC frameworks rely on iterative optimization, requiring frequent exchanges of high-dimensional features or similarity matrices between clients and the server. As the dataset size and view dimensions increase, the communication cost scales poorly, becoming a significant bottleneck. While feature compression mitigates this issue, it often causes semantic information loss that harms clustering accuracy. (2) **Insufficient privacy protection.** Intermediate representations, such as embeddings or similarity matrices, still encode latent semantic structures, which can be exploited via reconstruction or inference attacks. Although some works transmit pseudo-labels or compressed features to replace raw data, privacy risks may accumulate over multiple rounds of communication. (3) **Lack of adaptive aggregation.** Server-side aggregation often adopts fixed or uniform weighting, which ignores the variations in data quality across clients. When noisy views are present, these static strategies significantly degrade clustering performance.

To address the above challenges, we propose a trusted federated multi-view clustering framework, termed EvoFMVC, which leverages evidence-based representation exchange (Liang et al. 2025d; Lu et al. 2025) and adaptive aggregation (Liang et al. 2025b,a) to achieve privacy-preserving and communication-efficient learning. Fig. 1 provides a conceptual comparison between existing FMVC approaches and our proposed EvoFMVC framework.

The main contributions of this paper are as follows:

- Instead of transmitting high-dimensional model parameters or similarity matrices, EvoFMVC employs lightweight *trusted evidence* as a compact medium for inter-client communication. This greatly reduces communication overhead without compromising semantic expressiveness.
- The trusted evidence encodes cluster probability distributions, avoiding explicit exposure of feature structures or

latent embeddings, thereby reducing the risk of inversion or inference attacks over multiple rounds.

- EvoFMVC formulates the server-side fusion as a neural architecture search (NAS) task, enabling the server to automatically select and compose fusion operators via evolutionary algorithms, which enhances robustness to noisy views and boosts global clustering quality.

2 Related Work

In this section, we briefly review the current research status of federated learning and federated multi-view clustering.

2.1 Federated Learning

Federated learning (FL) is a privacy-preserving distributed learning paradigm that enables multiple clients or devices to collaboratively construct a global model while keeping all private data local (Konečný et al. 2017; McMahan et al. 2017; Shen et al. 2025). Depending on how data is distributed among clients, FL can be categorized into horizontal, vertical and transfer-based settings (Zhu et al. 2021). In horizontal FL (HFL), clients share the same feature space but hold different samples (McMahan et al. 2017), whereas vertical FL (VFL) involves clients with different feature spaces but aligned sample identifiers (Liu et al. 2024), making it suitable for cross-organization collaboration with feature-level privacy preservation. In contrast, federated transfer learning (FTL) deals with heterogeneous clients that differ in both sample and feature spaces, and thus relies on transfer learning to enable cross-domain knowledge sharing (Liu et al. 2020).

2.2 Federated Multi-View Clustering

Federated multi-view clustering aims to cluster view data distributed across different devices in a decentralized manner. The main challenge lies in effectively aggregating latent representations from multiple clients without direct access to their raw view data, under the constraints of privacy preservation and communication efficiency. For instance, Chen et al. (Chen et al. 2023) proposed FedMVL, which aligns multi-view features into a shared latent space via orthogonal matrix factorization. This method incorporates structural compression to reduce communication overhead during federated training and fuses information from different views using static weighting strategies. Mu et al. (Mu et al. 2024) proposed the FedDMC method, which utilizes deep autoencoders on the client side to perform dimensionality reduction and generate sample-level clustering probability distributions as pseudo-labels. These pseudo-labels, along with latent representations, are transmitted to the server for global clustering coordination. Feng et al. (Feng et al. 2024) proposed TensorFMVC, a tensor factorization-based framework that avoids centroid initialization by adopting centerless K-means and stacks local clustering assignments into a third-order tensor regularized by the Schatten-p norm. Similarly, Li et al. (Li, Yang, and Xie 2025) introduced FedMSGL, which learns self-expressive subspace representations locally and constructs a hypergraph on the server side to perform spectral clustering. Sun et al. (Sun et al.

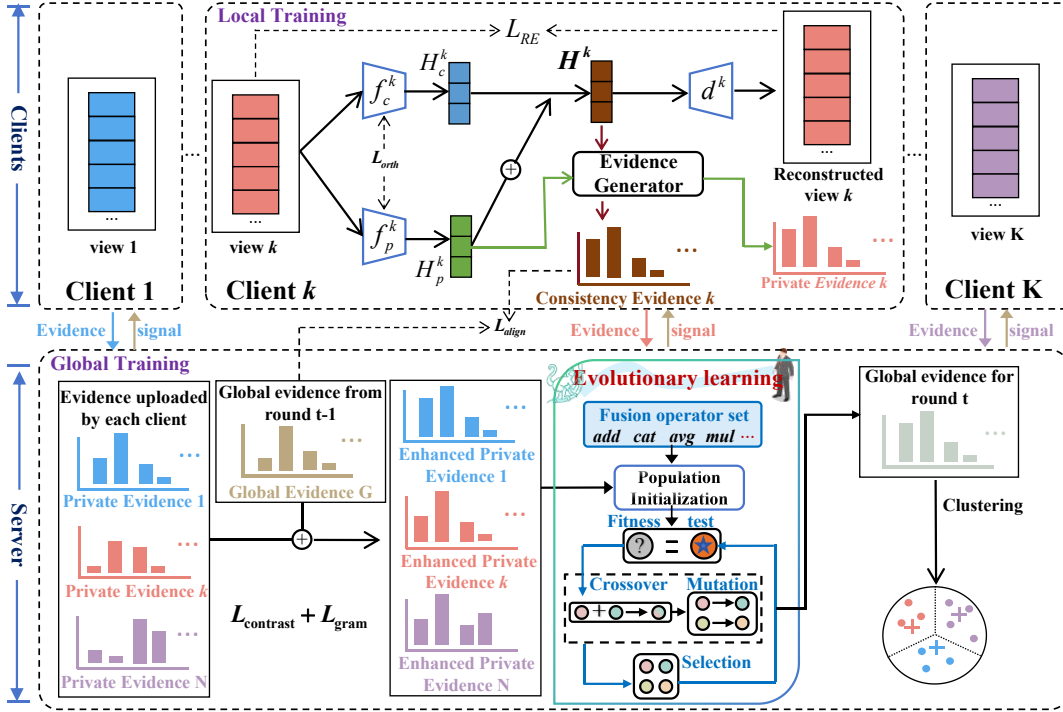


Figure 2: The whole framework of the proposed method.

2025) proposed a graph-based federated multi-view clustering method that jointly models cross-view consistency and view-specific diversity. Clients construct local graphs, while the server builds a global consistency graph and measures diversity to retain unique view characteristics.

3 Methodology

3.1 Problem Definition

Federated Multi-View Clustering (FMVC) aims to learn a global clustering model from data distributed across multiple clients, where each client holds a distinct view of a shared set of samples, with all views aligned across clients at the sample level. In a typical FMVC setting, there are K distinct views, each associated with a dedicated local client. These clients do not share raw data, ensuring privacy and data isolation across views. Given a multi-view dataset $\mathcal{X} = \{\mathbf{X}^{(1)}, \dots, \mathbf{X}^{(K)}\}$, where $\mathbf{X}^{(k)} \in \mathbb{R}^{N \times d_k}$ denotes the data held by the k -th client, we define the i -th sample from view k as $\mathbf{x}_i^{(k)} \in \mathbb{R}^{d_k}$. We assume all clients share a common set of N aligned samples, such that $\{\mathbf{x}_i^{(1)}, \dots, \mathbf{x}_i^{(K)}\}$ represent different views of the same sample.

In this paper, we propose a novel evidence-based FMVC method shown in Fig. 2 EvoFMVC leverages both the global evidence aggregated at the server and the private evidence provided by each client to learn high-quality clustering representations, based on which the server performs the final clustering. Next, we describe our method following the standard federated learning workflow, dividing the process into two parts: Local Training and Global Training.

3.2 Local Training

In each client k , corresponding to the k -th view, we train a local model to extract latent representations and generate view-specific evidence signals from its local data $\mathbf{X}^{(k)}$. To model shared and private semantics, we define two encoders on each client: a consistency encoder $f_c^k(\cdot)$ and a private encoder $f_p^k(\cdot)$, along with a decoder $d^k(\cdot)$. For each sample $\mathbf{x}_i^{(k)}$, the two encoders produce:

$$\mathbf{h}_{c,i}^k = f_c^k(\mathbf{x}_i^{(k)}), \quad \mathbf{h}_{p,i}^k = f_p^k(\mathbf{x}_i^{(k)}), \quad (1)$$

here, $\mathbf{h}_{c,i}^k$ and $\mathbf{h}_{p,i}^k$ denote the consistency and private latent representations of the i -th sample in client k , respectively. To encourage the two encoders to extract non-overlapping semantics (Lin et al. 2025), we introduce an orthogonality loss that penalizes their alignment:

$$\mathcal{L}_{\text{orth}}^{(k)} = \frac{1}{N} \sum_{i=1}^N \left((\mathbf{h}_{c,i}^k)^\top \mathbf{h}_{p,i}^k \right)^2. \quad (2)$$

To encourage private encoder to retain sufficient information for accurate reconstruction, we define the following loss:

$$\mathcal{L}_{\text{re}}^{(k)} = \frac{1}{N} \sum_{i=1}^N \left\| d^k(\mathbf{h}_{p,i}^k) - \mathbf{x}_i^{(k)} \right\|_2^2. \quad (3)$$

To transform the latent representations into a form suitable for evidential reasoning, we apply the Softplus activation to the outputs of both encoders. This activation ensures

that the resulting evidence vectors are non-negative, representing the degree of support for each category. Specifically, the evidence vectors are computed as:

$$\mathbf{e}_{c,i}^k = \text{Softplus}(\text{Linear}(\mathbf{h}_{c,i}^k)), \quad (4)$$

$$\mathbf{e}_{p,i}^k = \text{Softplus}(\text{Linear}(\mathbf{h}_{p,i}^k)). \quad (5)$$

The Softplus activation ensures that the transformed features are smooth and non-negative, enabling them to be interpreted as evidence vectors that reflect the model’s degree of support for each category. However, these evidence values cannot directly represent a probability distribution. To model categorical uncertainty in a principled way, we adopt the Dirichlet distribution, whose parameters for each sample are derived by shifting the evidence vectors by one:

$$\boldsymbol{\alpha}_{c,i}^{(k)} = \mathbf{e}_{c,i}^k + 1, \quad (6)$$

$$\boldsymbol{\alpha}_{p,i}^{(k)} = \mathbf{e}_{p,i}^k + 1, \quad (7)$$

where $\boldsymbol{\alpha}_{c,i}^{(k)}$ and $\boldsymbol{\alpha}_{p,i}^{(k)}$ are the Dirichlet parameters derived from consistency and private evidence, respectively, for the i -th sample in client k .

Following the previous round’s aggregation, the server computes a global Dirichlet prototype $\boldsymbol{\alpha}_g^{(t-1)}$ and broadcasts it to all clients. To enforce semantic alignment between local Dirichlet parameters and the global prototype, we define the following sample-wise alignment loss:

$$\mathcal{L}_{\text{align}} = \frac{1}{NK} \sum_{k=1}^K \sum_{i=1}^N \left\| \boldsymbol{\alpha}_{c,i}^{(k)} - \boldsymbol{\alpha}_g^{(t-1)} \right\|_2^2. \quad (8)$$

3.3 Global Training

At each communication round, the server receives Dirichlet parameters from all clients. For each client k , each sample i is represented by a consistency-based Dirichlet vector $\boldsymbol{\alpha}_{c,i}^{(k)} \in \mathbb{R}^C$, where C is the number of clusters. These vectors reflect the client’s belief distribution over latent cluster assignments.

To improve the semantic consistency and information quality of these local evidences, the server integrates the global prototype $\boldsymbol{\alpha}_g^{(t-1)} \in \mathbb{R}^C$, obtained in the previous round, into each local evidence vector by a weighted fusion:

$$\tilde{\boldsymbol{\alpha}}_i^{(k)} = \lambda \cdot \boldsymbol{\alpha}_{c,i}^{(k)} + (1 - \lambda) \cdot \boldsymbol{\alpha}_g^{(t-1)}, \quad (9)$$

where $\lambda \in [0, 1]$ is a hyperparameter controlling the influence of the global prototype. The resulting enhanced parameters $\tilde{\boldsymbol{\alpha}}_i^{(k)}$ are then used in downstream aggregation, guided by an evolutionary algorithm that adaptively selects high-quality client views and optimizes fusion operators. These enhanced parameters are candidates for global fusion. However, directly averaging them may lead to suboptimal results due to varying evidence quality across clients. To address this, we defer the final fusion process to an evolutionary algorithm, which adaptively selects a subset of high-quality clients and searches for optimal fusion strategies. The full design of this procedure is described in Section 3.5. After

Algorithm 1: Training Procedure of EvoFMVC

Require: Multi-view dataset $\mathcal{X} = \{\mathbf{X}^{(1)}, \dots, \mathbf{X}^{(K)}\}$; number of rounds T ;

Ensure: Final global Dirichlet parameters $\boldsymbol{\alpha}_g^{(T)}$ for clustering

- 1: **Initialize:** Server sets initial global prototype $\boldsymbol{\alpha}_g^{(0)}$
 - 2: **for** each round $t = 1$ to T **do**
 - 3: **for** each client $k = 1$ to K **in parallel do**
 - 4: Receive $\boldsymbol{\alpha}_g^{(t-1)}$ from server
 - 5: **for** each sample $\mathbf{x}_i^{(k)}$ **do**
 - 6: Encode with consistency and privacy encoders: $\mathbf{h}_{c,i}^k, \mathbf{h}_{p,i}^k$
 - 7: Compute $\mathcal{L}_{\text{orth}}^{(k)}, \mathcal{L}_{\text{re}}^{(k)}$ (Eqs. 2,3)
 - 8: Obtain evidences $\mathbf{e}_{c,i}^k, \mathbf{e}_{p,i}^k$ via Evidence Generator (Eq. 4,5)
 - 9: Compute Dirichlet: $\boldsymbol{\alpha}_{c,i}^{(k)} = \mathbf{e}_{c,i}^k + 1, \boldsymbol{\alpha}_{p,i}^{(k)} = \mathbf{e}_{p,i}^k + 1$
 - 10: **end for**
 - 11: Compute alignment loss $\mathcal{L}_{\text{align}}^{(k)}$ (Eq. 8)
 - 12: Upload private evidences $\{\boldsymbol{\alpha}_{p,i}^{(k)}\}_{i=1}^N$ to server
 - 13: **end for**
 - 14: **Server:** Fuse each $\boldsymbol{\alpha}_{p,i}^{(k)}$ with prototype: $\tilde{\boldsymbol{\alpha}}_i^{(k)} \leftarrow \lambda \boldsymbol{\alpha}_{p,i}^{(k)} + (1 - \lambda) \boldsymbol{\alpha}_g^{(t-1)}$
 - 15: Select high-quality clients and fusion operators via evolutionary algorithm (see Section 3.5)
 - 16: Aggregate selected $\tilde{\boldsymbol{\alpha}}_i^{(k)}$ into global evidence $\boldsymbol{\alpha}_g^{(t)}$
 - 17: Compute $\mathcal{L}_{\text{contrast}}$ and $\mathcal{L}_{\text{gram}}$ based on filtered confident samples (Eqs. 10–14)
 - 18: Compute total global loss: $\mathcal{L}_{\text{global}} = \lambda_{\text{contrast}} \cdot \mathcal{L}_{\text{contrast}} + \lambda_{\text{gram}} \cdot \mathcal{L}_{\text{gram}}$
 - 19: **end for**
 - 20: **Return:** Final global parameters $\boldsymbol{\alpha}_g^{(T)}$ for k-means clustering
-

fusion, the server optimizes the global evidence representations using two complementary objectives: contrastive alignment and structural diversity regularization.

To promote semantic consistency among high-quality clients, we adopt a contrastive loss aligning the enhanced evidences of the same sample across selected clients. Instead of relying on raw Dirichlet vectors, we derive belief distributions and use uncertainty-aware pseudo labels to filter out ambiguous samples. For each sample i in client v , compute

$$\mathbf{b}_i^{(v)} = \frac{\tilde{\boldsymbol{\alpha}}_i^{(v)} - 1}{\sum_{c=1}^C \tilde{\alpha}_{i,c}^{(v)}}, \quad u_i^{(v)} = \frac{C}{\sum_{c=1}^C \tilde{\alpha}_{i,c}^{(v)}}, \quad (10)$$

$$\hat{y}_i^{(v)} = \arg \max_c \mathbf{b}_{i,c}^{(v)}.$$

where $u_i^{(v)}$ is the uncertainty score, and $\hat{y}_i^{(v)}$ is the pseudo label. We define a binary mask $\mathbf{m}_i^{(v)} = \mathbb{I}[u_i^{(v)} < \epsilon]$ to select confident samples. Let $\mathcal{P} = \{(v, w) \mid v \neq w, v, w \in \mathcal{S}\}$ be the unordered client pair set based on selected clients from

the evolutionary fusion structure. To simplify the contrastive formulation, we define a soft similarity ratio between positive and negative pairs as follows:

$$\mathcal{R}_i^{(v,w)} = \log \frac{\sum_{j=1, j \neq i, \hat{y}_j^{(w)} = \hat{y}_i^{(v)}} \exp\left(\frac{\mathbf{b}_i^{(v)} \cdot \mathbf{b}_j^{(w)}}{\tau}\right)}{\sum_{j=1, j \neq i} \exp\left(\frac{\mathbf{b}_i^{(v)} \cdot \mathbf{b}_j^{(w)}}{\tau}\right)}, \quad (11)$$

$$\mathcal{L}_{\text{contrast}} = \frac{1}{|\mathcal{P}|N} \sum_{(v,w) \in \mathcal{P}} \sum_{i=1}^N \mathbf{m}_i^{(v)} \cdot \mathbf{m}_i^{(w)} \cdot \mathcal{R}_i^{(v,w)}. \quad (12)$$

To encourage structural diversity across selected client views, we define a Gram-based regularization loss that penalizes relational similarity between views. For each selected client $v \in \mathcal{S}$, we extract the evidence matrix $E^{(v)} \in \mathbb{R}^{N \times C}$, composed of confident samples (filtered by uncertainty as in the contrastive loss) (Wang et al. 2025). We compute the sample-wise similarity (Gram) matrix as:

$$G^{(v)} = E^{(v)}(E^{(v)})^\top. \quad (13)$$

The Gram diversity loss is then formulated as:

$$\mathcal{L}_{\text{gram}} = \sum_{(v,w) \in \mathcal{P}} \text{Tr}(G^{(v)}G^{(w)}), \quad (14)$$

where $\mathcal{P} = \{(v,w) \mid v \neq w, v, w \in \mathcal{S}\}$ is the set of un-ordered selected client pairs.

This formulation encourages diverse structural relationships among selected high-confidence client views by penalizing pairwise alignment of sample-level similarity patterns.

The total loss is defined as:

$$\mathcal{L}_{\text{global}} = \lambda_{\text{contrast}} \cdot \mathcal{L}_{\text{contrast}} + \lambda_{\text{gram}} \cdot \mathcal{L}_{\text{gram}}, \quad (15)$$

where λ_{gram} is a hyperparameter that balances the contribution of structural diversity against alignment.

3.4 Optimization

Alg. 1 summarizes the optimization process of EvoFMVC, which consists of two main components: local clients and the central server. In each communication round t , the server first distributes the global Dirichlet prototype $\alpha_g^{(t-1)}$ to all clients. Each client then performs local training, where consistency and private features are extracted, evidence vectors are computed, and Dirichlet parameters $\alpha_{c,i}^{(k)}, \alpha_{p,i}^{(k)}$ are derived. The clients also compute local alignment losses and upload their updated Dirichlet evidences to the server. Upon receiving the evidences, the server fuses each with the global prototype to obtain enhanced representations. These are passed to the evolutionary algorithm, which selects high-quality clients and optimizes fusion operators for global aggregation (Fu et al. 2024, 2025; Jin et al. 2025). After fusion, the server refines the global representations via two losses: a contrastive loss to promote alignment across views and a Gram-based regularization to preserve structural diversity (Liang et al. 2025c). This process is repeated for T communication rounds. After convergence, the final global Dirichlet parameters $\alpha_g^{(T)}$ are directly clustered in the Dirichlet parameter space using standard k-means, yielding the final clustering assignments.

3.5 Evolutionary View Fusion Strategy

In multi-view learning, traditional fusion methods often rely on fixed designed strategies, which may struggle with view heterogeneity. In contrast, evolutionary algorithms (EAs) provide structural adaptivity and search diversity, making them more suitable for complex aggregation scenarios (Liang et al. 2024, 2021; Guo et al. 2025; Cui et al. 2025). To fuse information from heterogeneous clients, we propose an EA to adaptively select informative clients and optimal fusion operations. It aims to search over possible fusion structures and improve global clustering performance.

Encoding: Each individual in the population is encoded as a variable-length vector composed of two parts: a selected client (view) index sequence $\mathbf{v} = [v_1, \dots, v_m]$, and a corresponding operator sequence $\mathbf{f} = [f_1, \dots, f_{m-1}]$, where each operator represents a predefined fusion function (this paper uses addition, concatenation, multiplication, max, average). The full encoding is $p = [\mathbf{v}, \mathbf{f}]$.

Initialization: A population of size P is initialized by randomly generating multiple individuals with randomly sampled clients and operators.

Fitness Evaluation: For each individual, we first decode it into a specific deep fusion model; then train the model on the aggregated local evidence representations; finally evaluate clustering accuracy on a validation set as its fitness.

Crossover: Given two parent individuals $p^1 = [\mathbf{v}^1, \mathbf{f}^1]$ and $p^2 = [\mathbf{v}^2, \mathbf{f}^2]$, we apply single-point crossover separately on view and operator sequences to produce offspring.

Repair: After crossover, we ensure the length constraint $|\mathbf{v}| = |\mathbf{f}| + 1$ is satisfied by adding or removing operator elements from the appropriate side.

Mutation: We randomly change a selected view or fusion operator in the encoding with a fixed mutation probability.

Selection: Binary tournament selection is used to retain better individuals into the next generation.

This evolutionary search process proceeds for T generations. The best-performing individual is selected as the final fusion structure for global evidence integration.

4 Experiments

4.1 Settings

Implementation Details. All experiments are conducted on a server equipped with an AMD EPYC 7763 CPU, an NVIDIA RTX 4090 GPU (24GB), and 512GB RAM. Each client employs a consistency encoder and a private encoder to extract latent features, followed by a shared decoder and an Evidence Generator that outputs Dirichlet parameters. We pretrain the consistency and privacy encoders on each client using the Adam optimizer with a learning rate of 0.0001 and a batch size of 128. In the subsequent federated training, clients extract evidential representations locally and transmit private evidence to the server. The server performs evolutionary fusion across selected views to construct global evidence, which is then redistributed to clients to enhance local training. This process forms a closed-loop optimization.

Datasets. To comprehensively evaluate the performance of the proposed method, we conduct experiments on seven widely-used multi-view datasets, including Scene (Fei-Fei

| Dataset | Metric | LMVSC | SiMVC | CoMVC | MFLVC | GCFAgg | FedMVL | FedDMVC | FCUIF | MGCD | Ours |
|-----------|--------|--------|--------|---------------|---------------|---------------|---------|---------------|---------------|---------------|---------------|
| Scene | ACC | 0.3222 | 0.4383 | 0.4347 | 0.3173 | 0.2022 | 0.0945 | 0.4360 | 0.4252 | <u>0.5285</u> | 0.7640 |
| | NMI | 0.3396 | 0.4657 | 0.4627 | 0.3392 | 0.1842 | 0.0100 | 0.4184 | 0.3880 | <u>0.5625</u> | 0.8151 |
| | ARI | 0.1714 | 0.2787 | 0.2710 | 0.1784 | 0.0746 | 0.0643 | 0.2697 | 0.2474 | <u>0.3665</u> | 0.6218 |
| | PUR | 0.3922 | 0.5084 | 0.5001 | 0.3456 | 0.2486 | 0.1064 | 0.4237 | 0.4020 | <u>0.5845</u> | 0.7640 |
| Aloi | ACC | 0.6390 | 0.6730 | 0.7010 | 0.7490 | 0.5745 | 0.0349 | 0.8566 | 0.7460 | 0.9160 | 0.9158 |
| | NMI | 0.7700 | 0.8530 | 0.8940 | 0.8570 | 0.8268 | 0.0731 | 0.9210 | 0.8426 | <u>0.9668</u> | 0.9735 |
| | ARI | 0.5030 | 0.5550 | 0.6530 | 0.6680 | 0.5184 | 0.0374 | 0.8050 | 0.6285 | <u>0.8992</u> | 0.9074 |
| | PUR | 0.6900 | 0.8150 | 0.7940 | 0.7810 | 0.5968 | 0.0361 | 0.8941 | 0.7926 | <u>0.9302</u> | 0.9328 |
| Animal | ACC | 0.1310 | 0.1600 | 0.1560 | 0.1910 | 0.1528 | 0.0791 | 0.1829 | 0.1793 | <u>0.5201</u> | 0.6631 |
| | NMI | 0.0290 | 0.1360 | 0.1350 | 0.1660 | 0.1480 | 0.0147 | 0.1641 | 0.1590 | <u>0.6491</u> | 0.8099 |
| | ARI | 0.0290 | 0.0530 | 0.0500 | 0.0750 | 0.0639 | 0.0125 | 0.0694 | 0.0665 | <u>0.4010</u> | 0.6062 |
| | PUR | 0.1390 | 0.1720 | 0.1640 | 0.2030 | 0.1929 | 0.1010 | 0.1720 | 0.1673 | <u>0.5715</u> | 0.7359 |
| Yale | ACC | 0.5333 | 0.5454 | 0.5636 | 0.2545 | 0.3273 | 0.2115 | <u>0.6061</u> | 0.5273 | 0.5125 | 0.6437 |
| | NMI | 0.5944 | 0.6092 | <u>0.6308</u> | 0.2407 | 0.3372 | 0.0043 | 0.3816 | 0.5965 | 0.6219 | 0.7809 |
| | ARI | 0.2986 | 0.3674 | 0.4031 | 0.0671 | 0.3394 | 0.0011 | 0.6933 | 0.3502 | 0.3411 | <u>0.5530</u> |
| | PUR | 0.5697 | 0.5515 | 0.5696 | 0.2545 | 0.0855 | 0.5384 | <u>0.6121</u> | 0.5455 | 0.5437 | 0.6625 |
| 100Leaves | ACC | 0.5419 | 0.5168 | 0.5350 | 0.3875 | <u>0.8306</u> | 0.1105 | 0.3994 | 0.6456 | 0.7839 | 0.8489 |
| | NMI | 0.7760 | 0.7875 | 0.7933 | 0.7081 | <u>0.9446</u> | 0.0378 | 0.2240 | 0.7990 | 0.8919 | 0.9475 |
| | ARI | 0.4029 | 0.4234 | 0.4249 | 0.2711 | <u>0.8015</u> | 0.0323 | 0.7236 | 0.4834 | 0.6538 | 0.8250 |
| | PUR | 0.5850 | 0.5381 | 0.5562 | 0.4031 | 0.7984 | 0.5961 | 0.4138 | 0.6656 | <u>0.8288</u> | 0.8873 |
| BBCSports | ACC | 0.6379 | 0.4485 | 0.3713 | <u>0.7022</u> | 0.5294 | 0.3028 | 0.6949 | 0.6488 | 0.6379 | 0.7536 |
| | NMI | 0.4773 | 0.1959 | 0.1489 | 0.5256 | 0.3417 | 0.0016 | 0.3969 | 0.6990 | 0.6074 | <u>0.6653</u> |
| | ARI | 0.2803 | 0.1541 | 0.0936 | 0.4492 | 0.5699 | -0.0002 | 0.5179 | 0.4897 | 0.5166 | <u>0.5212</u> |
| | PUR | 0.6618 | 0.5110 | 0.4117 | 0.7390 | 0.2630 | 0.5336 | 0.6949 | 0.6619 | <u>0.7261</u> | 0.7536 |
| NUS | ACC | 0.2558 | 0.2650 | 0.2550 | 0.2575 | 0.2296 | 0.1170 | <u>0.4575</u> | 0.2179 | 0.4019 | 0.5869 |
| | NMI | 0.1215 | 0.1473 | 0.1521 | 0.1366 | 0.1221 | 0.0118 | 0.1928 | 0.0878 | <u>0.3984</u> | 0.7622 |
| | ARI | 0.0561 | 0.0763 | 0.0756 | 0.0763 | 0.2396 | 0.0006 | <u>0.2616</u> | 0.0443 | 0.2239 | 0.5366 |
| | PUR | 0.2233 | 0.2687 | 0.2800 | 0.2708 | 0.0639 | 0.1191 | <u>0.4575</u> | 0.2237 | 0.4167 | 0.6613 |

Table 1: Comparative results between EvoFMVC and state-of-the-art methods on different datasets.

| Data | Samples | Clusters | View dimensions |
|-----------|---------|----------|-----------------------|
| Scene | 4485 | 15 | 20/59/40 |
| Aloi | 10800 | 100 | 77/13/64/125 |
| Animal | 11673 | 20 | 2689/2000/2001/2000 |
| Yale | 165 | 15 | 4096/4096/4096 |
| 100Leaves | 1600 | 100 | 64/64/64 |
| BBCSport | 544 | 5 | 3183/3183 |
| NUS | 2400 | 12 | 64/144/73/128/225/500 |

Table 2: Statistical characteristics of the adopted datasets.

and Perona 2005), Aloi (Li et al. 2023), Animal (Li et al. 2016), Yale, 100Leaves, BBCSport (Greene and Cunningham 2006), and NUS (Chua et al. 2009). They vary significantly in terms of sample size, cluster numbers, and view dimensionalities, ensuring a diverse and representative evaluation across different multi-view clustering scenarios. Their statistical characteristics are summarized in Table 2.

The Compared Methods. To validate the effectiveness of EvoFMVC, we compare it with eight state-of-the-art multi-

view clustering baselines, including five centralized methods and three federated methods. The centralized methods include LMVSC (Kang et al. 2020), SiMVC (Trosten et al. 2021), CoMVC (Trosten et al. 2021), MFLVC (Xu et al. 2022), and GCFAgg (Yan et al. 2023). The federated methods include FedMVL (Huang et al. 2022), FedDMVC (Chen et al. 2023), and FCUIF (Ren et al. 2024). They are implemented based on the official codes provided by the respective authors. The hyperparameters are tuned according to the suggestions in the original literature to ensure fair and competitive evaluations.

Metrics. Following common practice in multi-view clustering research (Liang et al. 2022), we evaluate performance using four metrics: Accuracy (ACC), Normalized Mutual Information (NMI), Purity (PUR), and Adjusted Rand Index (ARI). Higher values indicate better performance.

4.2 Comparable Results

Table 1 records the experimental comparison of EvoFMVC with eight comparison methods on eight datasets, where the best and second-best performances are highlighted in bold

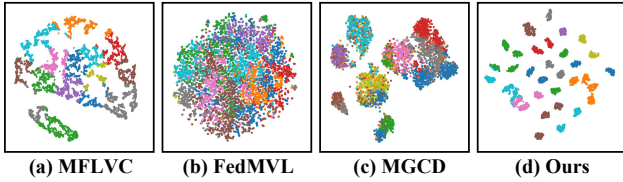


Figure 3: Comparison of clustering results.

| Dataset | Method | <i>Dim.</i> | ACC | NMI | ARI | PUR |
|------------|--------|-------------|---------------|---------------|---------------|---------------|
| 1000Leaves | MGCD | 512 | 0.7839 | 0.8919 | 0.6538 | 0.8288 |
| | Ours | 100 | 0.8489 | 0.9475 | 0.8250 | 0.8873 |
| BBCSports | MGCD | 512 | 0.6379 | 0.6074 | 0.5166 | 0.7261 |
| | Ours | 5 | 0.7536 | 0.6653 | 0.5212 | 0.7536 |

Table 3: Comparison of the two methods under their respective optimal performance. *Dim.* indicates the model dimension required to achieve the best performance (higher dimensions mean higher computational cost).

and underlined, respectively. According to Table 1, we can observe that: Across all datasets presented in the table, our proposed EvoFMVC method consistently outperforms all baseline methods across all evaluation metrics. Notably, on the Scene and NUS datasets, EvoFMVC achieves significant improvements over the second-best methods. For the Scene dataset, the relative gains are 23.55%, 25.26%, 25.53%, and 17.95% in ACC, NMI, ARI, and PUR, respectively. For the NUS dataset, EvoFMVC obtains relative improvements of 18.5%, 36.28%, 31.27%, and 24.46% in the same four metrics. These experimental results clearly demonstrate the effectiveness and robustness of our approach.

As shown in Fig. 3, we provide visual comparisons between EvoFMVC and four representative federated multi-view clustering methods. The results clearly demonstrate that EvoFMVC yields more accurate and visually separable clusters, reflecting its strong overall clustering performance.

Beyond clustering accuracy, we also examine the efficiency of each method in terms of the feature dimensionality required to achieve optimal performance. As shown in Table 3, EvoFMVC achieves competitive results with significantly lower feature dimensions compared to MGCD. For instance, on the 1000Leaves dataset, EvoFMVC reaches its best performance with only 100-dimensional representations, whereas MGCD requires 512 dimensions. A similar trend is observed on the BBCSports dataset. This suggests that EvoFMVC is more parameter-efficient and computationally lightweight, making it suitable for deployment in resource-constrained federated environments.

To further evaluate method performance under consistent efficiency constraints, we conduct comparisons in a unified low-dimensional setting. Specifically, the feature dimensionality is set to match the number of classes for each dataset, which provides a minimal yet fair representation. As shown in Table 4, even under this compact representation setting, EvoFMVC still outperforms MGCD across all datasets. These results confirm the robustness and adaptabil-

ity of our method under strict computational limitations.

| Dataset | <i>Dim.</i> | Method | ACC | NMI | ARI | PUR |
|---------|-------------|--------|---------------|---------------|---------------|---------------|
| Scene | 15 | MGCD | 0.3545 | 0.3611 | 0.1819 | 0.3766 |
| | | Ours | 0.7640 | 0.8151 | 0.6218 | 0.7640 |
| aloi | 100 | MGCD | 0.7142 | 0.8779 | 0.6425 | 0.7623 |
| | | Ours | 0.9158 | 0.9735 | 0.9074 | 0.9328 |
| Yale | 15 | MGCD | 0.3812 | 0.4984 | 0.2128 | 0.4000 |
| | | Ours | 0.6437 | 0.7809 | 0.5530 | 0.6625 |

Table 4: The performance advantage of our methods under the same efficiency.

4.3 Ablation Results

To verify the effectiveness of our proposed adaptive evidence fusion strategy, we conduct an ablation study (see Table 5) by replacing it with a naive baseline that directly sums the Dirichlet evidence received from all local clients. This naive summation assumes equal importance across all views without considering their individual quality or contribution. In contrast, our approach employs an evolution-based optimization strategy to adaptively weight and fuse the evidence, aiming to better capture complementary information among views. Experimental results show that the adaptive fusion consistently outperforms simple summation in terms of clustering accuracy and robustness, demonstrating the superiority of our strategy in federated multi-view settings.

| Dataset | Method | ACC | NMI | ARI | PUR |
|----------|------------------|---------------|---------------|---------------|---------------|
| Scene | Naive Sum | 0.6855 | 0.7760 | 0.5556 | 0.7337 |
| | EvoFusion | 0.7640 | 0.8151 | 0.6218 | 0.7640 |
| BBCSport | Naive Sum | 0.5974 | 0.6246 | 0.3825 | 0.7022 |
| | EvoFusion | 0.7536 | 0.6653 | 0.5212 | 0.7536 |

Table 5: Ablation study on the evidence fusion strategy.

5 Conclusion

This paper presents a trusted federated multi-view clustering framework EvoFMVC, which incorporates an evolutionary fusion Algorithm. In this framework, trusted evidence is employed as the medium for information exchange between the server and the clients, thereby ensuring data privacy and security while significantly reducing communication overhead. On the server side, an evolutionary fusion algorithm integrates the multi-view information uploaded by individual clients to construct more reliable global trusted evidence, which in turn enhances clustering performance. Extensive experiments conducted on multiple benchmark datasets demonstrate the effectiveness and superiority of the proposed method in various clustering tasks.

Acknowledgements

This work was supported by National Natural Science Foundation of China (Nos. 62306171, 62406218), the Science and Technology Major Project of Shanxi (No. 202201020101006), and the Self-initiated Research Project of the Foundation of Shanxi Key Laboratory of Big Data Analysis and Parallel Computing (No. BDPC-ZZ-23-001).

References

- Chen, X.; Xu, J.; Ren, Y.; Pu, X.; Zhu, C.; Zhu, X.; Hao, Z.; and He, L. 2023. Federated deep multi-view clustering with global self-supervision. In *Proceedings of the 31st ACM International Conference on Multimedia*, 3498–3506.
- Chua, T.-S.; Tang, J.; Hong, R.; Li, H.; Luo, Z.; and Zheng, Y. 2009. NUS-WIDE: a real-world web image database from National University of Singapore. In *Proceedings of the ACM International Conference on Image and Video Retrieval*.
- Cui, Z.; Sun, S.; Guo, Q.; Liang, X.; Qian, Y.; and Zhang, Z. 2025. A Fast Neural Architecture Search Method for Multi-Modal Classification via Knowledge Sharing. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, 5003–5011.
- Fang, X.; and Ye, M. 2025. Noise-Robust Federated Learning With Model Heterogeneous Clients. *IEEE Transactions on Mobile Computing*, 24(5): 4053–4071.
- Fei-Fei, L.; and Perona, P. 2005. A bayesian hierarchical model for learning natural scene categories. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, 524–531.
- Feng, W.; Wu, Z.; Wang, Q.; Dong, B.; Tao, Z.; and Gao, Q. 2024. Federated Multi-View Clustering via Tensor Factorization. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, 3962–3970.
- Fu, P.; Liang, X.; Luo, T.; Guo, Q.; Zhang, Y.; and Qian, Y. 2024. Core-Structures-Guided Multi-Modal Classification Neural Architecture Search. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, 3980–3988.
- Fu, P.; Liang, X.; Qian, Y.; Guo, Q.; Zhang, Y.; Huang, Q.; and Tang, K. 2025. Multi-Scale Features Are Effective for Multi-Modal Classification: An Architecture Search Viewpoint. *IEEE Transactions on Circuits and Systems for Video Technology*, 35(2): 1070–1083.
- Greene, D.; and Cunningham, P. 2006. Practical solutions to the problem of diagonal dominance in kernel document clustering. In *Proceedings of the 23rd International Conference on Machine Learning*, 377–384.
- Guo, Q.; Liang, S.; Sun, Xinyan; Qian, Y.; and Cui, Z. 2025. LogicNAS: Multi-view neural architecture search method for image sequence logic prediction. *IEEE Transactions on Emerging Topics in Computational Intelligence*.
- Huang, S.; Shi, W.; Xu, Z.; Tsang, I.; and Lv, J. 2022. Efficient federated multi-view learning. *Pattern Recognition*, 131: 108817.
- Huang, W.; Wan, G.; Ye, M.; and Du, B. 2023. Federated graph semantic and structural learning. *IJCAI '23*. ISBN 978-1-956792-03-4.
- Jin, Z.; Qian, Y.; Liang, X.; and Geng, H. 2025. A Multi-view Fusion Approach for Enhancing Speech Signals via Short-time Fractional Fourier Transform. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, 5508–5516.
- Kang, Z.; Zhou, W.; Zhao, Z.; Shao, J.; Han, M.; and Xu, Z. 2020. Large-scale multi-view subspace clustering in linear time. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 4412–4419.
- Konečný, J.; McMahan, H. B.; Yu, F. X.; Richtárik, P.; Suresh, A. T.; and Bacon, D. 2017. Federated Learning: Strategies for Improving Communication Efficiency. arXiv:1610.05492.
- Li, D.; Yang, Z.; and Xie, S. 2025. FedMSGL: A Self-Expressive Hypergraph Based Federated Multi-View Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 18244–18252.
- Li, X.; Ren, Z.; Sun, Q.; and Xu, Z. 2023. Auto-weighted tensor Schatten p-norm for robust multi-view graph clustering. *Pattern Recognition*, 134: 109083.
- Li, Y.; Shi, X.; Du, C.; Liu, Y.; and Wen, Y. 2016. Manifold regularized multi-view feature selection for social image annotation. *Neurocomputing*, 204: 135–141.
- Liang, X.; Fu, P.; Guo, Q.; Zheng, K.; and Qian, Y. 2024. DC-NAS: Divide-and-Conquer Neural Architecture Search for Multi-Modal Classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 13754–13762.
- Liang, X.; Fu, P.; Qian, Y.; Guo, Q.; and Liu, G. 2025a. Trusted multi-view classification via evolutionary multi-view fusion. In *Proceedings of the 13th International Conference on Learning Representations*, 1–14.
- Liang, X.; Guo, Q.; Qian, Y.; Ding, W.; and Zhang, Q. 2021. Evolutionary Deep Fusion Method and its Application in Chemical Structure Recognition. *IEEE Transactions on Evolutionary Computation*, 25(5): 883–893.
- Liang, X.; Li, S.; Guo, Q.; Qian, Y.; Jiang, B.; Luo, T.; and Du, L. 2025b. Evolutionary Multi-View Classification via Eliminating Individual Fitness Bias. In *Proceedings of the Thirty-ninth Annual Conference on Neural Information Processing Systems*.
- Liang, X.; Lv, L.; Guo, Q.; Jiang, B.; Li, F.; Du, L.; and Chen, L. 2025c. View-Association-Guided Dynamic Multi-View Classification. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, 5680–5688.
- Liang, X.; Wang, S.; Qian, Y.; Guo, Q.; Du, L.; Jiang, B.; Luo, T.; and Li, F. 2025d. Trusted Multi-View Classification with Expert Knowledge Constraints. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, 37409–37426.
- Liang, Y.; Huang, D.; Wang, C.-D.; and Yu, P. S. 2022. Multi-view graph learning by joint modeling of consistency

- and inconsistency. *IEEE Transactions on Neural Networks and Learning Systems*, 35(2): 2848–2862.
- Lin, Y.; Wang, Y.; Lyu, G.; Deng, Y.; Cai, H.; Lin, H.; Wang, H.; and Yang, Z. 2025. Enhance Multi-View Classification Through Multi-Scale Alignment and Expanded Boundary. In *Proceedings of the International Conference on Learning Representations, ICLR-2025*, 47604–47622.
- Liu, Y.; Kang, Y.; Xing, C.; Chen, T.; and Yang, Q. 2020. A Secure Federated Transfer Learning Framework. *IEEE Intelligent Systems*, 35(4): 70–82.
- Liu, Y.; Kang, Y.; Zou, T.; Pu, Y.; He, Y.; Ye, X.; Ouyang, Y.; Zhang, Y.-Q.; and Yang, Q. 2024. Vertical Federated Learning: Concepts, Advances, and Challenges. *IEEE Transactions on Knowledge and Data Engineering*, 36(7): 3615–3634.
- Lu, J.; Buntine, W.; Qi, Y.; Dipnall, J.; Gabbe, B.; and Du, L. 2025. Navigating Conflicting Views: Harnessing Trust for Learning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, 40411–40435.
- McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; and Arcas, B. A. y. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54, 1273–1282.
- Mu, X.; Cheng, K.; Shen, Y.; Li, X.; Chang, Z.; Zhang, T.; and Ma, X. 2024. FedDMC: Efficient and Robust Federated Learning via Detecting Malicious Clients. *IEEE Transactions on Dependable and Secure Computing*, 21(6): 5259–5274.
- Ren, Y.; Chen, X.; Xu, J.; Pu, J.; Huang, Y.; Pu, X.; Zhu, C.; Zhu, X.; Hao, Z.; and He, L. 2024. A novel federated multi-view clustering method for unaligned and incomplete data fusion. *Information Fusion*, 108: 102357.
- Shen, W.; Ye, M.; Yu, W.; and Yuen, P. C. 2025. Build Yourself Before Collaboration: Vertical Federated Learning With Limited Aligned Samples. *IEEE Transactions on Mobile Computing*, 24(7): 6503–6516.
- Sun, B.; Deng, Y.; Lin, Y.; Hai, Q.; Yang, Z.; and Lyu, G. 2025. Graph Consistency and Diversity Measurement for Federated Multi-View Clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 20663–20671.
- Trosten, D. J.; Lokse, S.; Jenssen, R.; and Kampffmeyer, M. 2021. Reconsidering representation alignment for multi-view clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1255–1265.
- Wang, Q.; Zhang, Z.; Feng, W.; Tao, Z.; and Gao, Q. 2025. Contrastive Multi-View Subspace Clustering via Tensor Transformers Autoencoder. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(20): 21207–21215.
- Xu, J.; Tang, H.; Ren, Y.; Peng, L.; Zhu, X.; and He, L. 2022. Multi-level feature learning for contrastive multi-view clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16051–16060.
- Yan, W.; Zhang, Y.; Lv, C.; Tang, C.; Yue, G.; Liao, L.; and Lin, W. 2023. Gcfagg: Global and cross-view feature aggregation for multi-view clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19863–19872.
- Zhu, H.; Xu, J.; Liu, S.; and Jin, Y. 2021. Federated learning on non-IID data: A survey. *Neurocomputing*, 465: 371–390.