An Effective and Secure Federated Multi-View Clustering Method with Information-Theoretic Perspective

Xinyue Chen [1] **Jinfeng Peng** [2] **Yuhao Li** [1] **Xiaorong Pu** [1 2] **Yang Yang** [1] **Yazhou Ren** [1 2]

Abstract

Recently, federated multi-view clustering (FedMVC) has gained attention for its ability to mine complementary clustering structures from multiple clients without exposing private data. Existing methods mainly focus on addressing the feature heterogeneity problem brought by views on different clients and mitigating it using shared client information. Although these methods have achieved performance improvements, the information they choose to share, such as model parameters or intermediate outputs, inevitably raises privacy concerns. In this paper, we propose an Effective and Secure Federated Multi-view Clustering method, ESFMC, to alleviate the dilemma between privacy protection and performance improvement. This method leverages the information-theoretic perspective to split the features extracted locally by clients, retaining sensitive information locally and only sharing features that are highly relevant to the task. This can be viewed as a form of privacy-preserving information sharing, reducing privacy risks for clients while ensuring that the server can mine high-quality global clustering structures. Theoretical analysis and extensive experiments demonstrate that the proposed method more effectively mitigates the trade-off between privacy protection and performance improvement compared to state-of-the-art methods.

1. Introduction

The advancement of technology and comprehensive attention to various matters have facilitated the emergence of multi-view data, which usually comes from multiple sources or perspectives (Fang et al., 2023; Huang et al., 2024; Xu et al., 2024; Pu et al., 2024). Multi-view clustering aims to explore and integrate multi-view data in an unsupervised manner and is applied in practical fields such as recommendation systems (Yu et al., 2018), social network analysis (Cruickshank, 2020), and bioinformatics (Lin et al., 2024). Most existing multi-view clustering methods usually assume that multi-view data are stored centrally in a single entity. However, due to the presence of data silos and the security issues associated with data exposure, unifying data from different sources incurs significant costs and privacy risks. Consequently, federated multi-view clustering (FedMVC) is gaining attention for its ability to mine complementary clustering structures from multiple clients without exposing private data (Lin et al., 2023; Qiao et al., 2023).

Existing FedMVC methods primarily address the issue of feature heterogeneity brought by views on different clients. For instance, federated deep multi-view clustering (Chen et al., 2023) mitigates the heterogeneity of local data by constructing global self-supervised information. Building on resolving heterogeneity issues, some works further explore incomplete data (Ren et al., 2024), communication variability (Huang et al., 2022), and Non-IID settings (Jiang et al., 2024). In these explorations, sharing client information is widely used as an effective means to alleviate data heterogeneity and improve method performance.

Although extensive works indicate that sharing certain information during client training can significantly enhance the clustering performance of FedMVC methods, it unfortunately raises privacy concerns. For example, researchers can reconstruct raw data from model parameters or shared gradients (Geiping et al., 2020), while model intermediate outputs may be vulnerable to model inversion attacks (Mahendran & Vedaldi, 2015). Techniques such as differential privacy offer provable security for privacy protection (Abadi et al., 2016), but the addition of random noise can negatively impact performance. Therefore, maintaining the performance of proposed methods while further protecting data privacy remains a critical challenge in the field of FedMVC.

We revisit the purpose of sharing information and propose an Effective and Secure Federated Multi-view Clustering

[1]School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China [2]Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, Shenzhen, China. Correspondence to: Yazhou Ren <yazhou.ren@uestc.edu.cn>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

method, ESFMC, to alleviate the dilemma between privacy protection and performance improvement. Specifically, ESFMC splits the features extracted locally by clients based on the information theory perspective, only sharing clustering-related features to the server while keeping sample-related features locally. This can be viewed as a form of privacy-preserving information sharing, where sensitive information in the features is retained locally, and only task-relevant features are shared. ESFMC minimizes the risk of privacy breaches by preventing malicious attackers from reconstructing the original data using the shared information. Additionally, ESFMC collaboratively aligns the non-overlapping samples across clients from the global perspective, effectively extending its applicability to incomplete scenarios. This approach reduces privacy risks for clients while ensuring that the server can mine complementary and high-quality global clustering structures. Our contributions are summarized as follows.

- We propose an effective and secure federated multi-view clustering method that aims to mine complementary global clustering structures while reducing privacy risks through limited information sharing.
- We design a collaborative alignment strategy that ensures the consistency of locally shared information across clients from the global perspective, thus extending the proposed method to incomplete scenarios.
- Theoretical analysis and extensive experiments demonstrate that the proposed method more effectively mitigates the trade-off between privacy protection and performance improvement compared to SOTAs.

2. Related Work

Federated multi-view clustering (FedMVC) is designed to handle multi-view data distributed across clients, ensuring data privacy while addressing the heterogeneity introduced by views on different clients through information sharing. Based on the type of information shared, FedMVC can be classified into two types. (1) Clients share information about local models, such as model parameters or shared gradients (Flanagan et al., 2021; Huang et al., 2022). (Che et al., 2022) aimed to improve local disease prediction performance by sharing training models among clients. (Jiang et al., 2024) introduced a heterogeneity-aware module that adapts FedMVC to IID and Non-IID scenarios by sharing model parameters and feature exchange. (Chen et al., 2024) addressed client and view gaps associated with heterogeneous hybrid views by sharing model parameters. (2) Clients share model intermediate outputs, such as embedded features or clustering assignments (Chen et al., 2023; Hu et al., 2023). By sharing features and global pseudo-labels, (Yan et al., 2024) designed a FedMVC strategy based on graph neural networks, addressing the issues of data privacy and feature heterogeneity. (Ren et al., 2024) introduced a FedMVC method, which addresses the issues of unaligned and incomplete data by sharing embedded features and designing adaptive alignment and imputation strategies.

The above methods achieve good performance by sharing model information or intermediate outputs. However, while these methods are successful, the information they share still leads to potential data leakage risks. For instance, attackers can reconstruct raw data from model parameters (Geiping et al., 2020) or features (Mahendran & Vedaldi, 2015). Thus, we propose an effective and secure federated multi-view clustering method to alleviate the dilemma between privacy protection and performance improvement.

3. Methodology

3.1. Motivation

Existing FedMVC methods aim to mine complementary clustering structures by sharing information such as model parameters (Che et al., 2022) and embedded features (Chen et al., 2023) from clients to the server. While this shared information significantly alleviates the heterogeneity issue caused by views on different clients, it often contains complete descriptions of clients' local data, retaining substantial critical information about the raw data. Attackers can easily reconstruct raw data using methods such as gradient-based attacks (Geiping et al., 2020) and model inversion attacks (Nguyen et al., 2023), leading to privacy risks.

Based on the intuitive idea that sharing partial information is more privacy-preserving than sharing complete information, we aim to split the information to maximize the utility of shared information. The information bottleneck principle (Tishby et al., 2000) suggests that ideal features should compress data while retaining as much task-relevant information as possible. This is achieved by minimizing the mutual information between $\mathbf{X}$ and $\mathbf{Z}$ while maximizing the mutual information between $\mathbf{Y}$ and $\mathbf{Z}$:

$$\mathcal{L}_{IB} = I(\mathbf{Z}; \mathbf{X}) - \beta I(\mathbf{Z}; \mathbf{Y}). \tag{1}$$

Inspired by Eq. (1) and previous works (Lee & Pavlovic, 2021; Yang et al., 2023), ESFMC splits the commonly used shared information in FedMVC, embedded features extracted from local clients, into clustering-related features and sample-related features. Clustering-related features focus on the similarities and differences among samples, facilitating the clustering process. Sample-related features focus on the properties and details of each individual sample. We consider that sample-related features contain too much private information and are irrelevant to the clustering task. Thus, clients retain sample-related features locally to prevent privacy leakage while sharing clustering-related

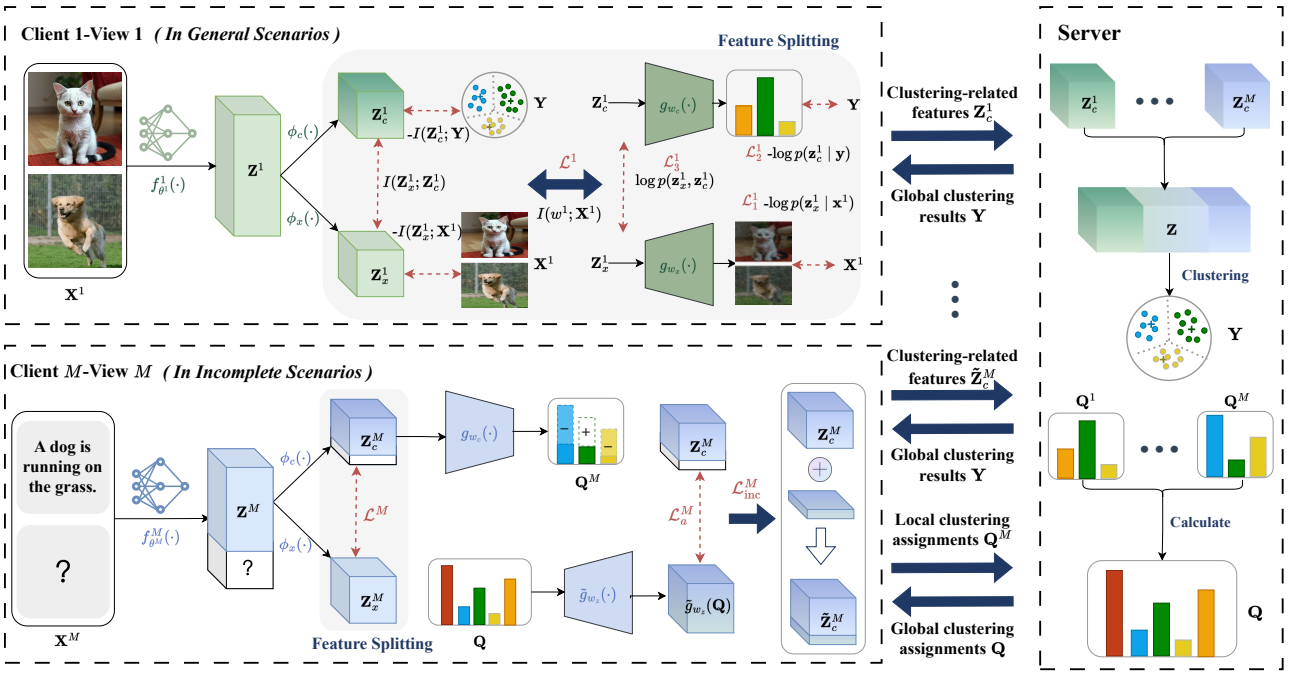


Figure 1. The framework of ESFMC. It contains M clients and a server. (1) ***In general scenarios:*** we propose feature splitting within clients from the information-theoretic perspective, where clustering-related features are shared to assist the server in mining the global clustering structures and enhancing performance, while sample-related features are retained locally to prevent privacy leakage. (2) ***In incomplete scenarios:*** we design a collaborative alignment strategy that leverages the server to integrate information and provide a global perspective, ensuring alignment among non-overlapping samples across clients.

features with the server, promoting the extraction of high-quality complementary clustering structures.

Additionally, inspired by (Hellström et al., 2023; Alquier et al., 2024), we introduce model-data mutual information $I(w; \mathbf{X})$ as a constraint to mitigate the model's dependency on data during local training. It quantifies the correlation between the model and input data, while also serving as a measure of model complexity for generalization analysis.

3.2. Problem Statement

Given multi-view data with M views, denoted by $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^M\}$, distributed among M clients, consisting of N samples, with the expectation of partitioning them into K clusters. For client m, its data are represented as $\mathbf{X}^m \in \mathbb{R}^{N_m \times D_m}$, where D_m is the dimensionality of samples in the m-th view and N_m is the number of samples in the m-th client. We assume that all participating parties are semi-honest and do not collude. The attacker follows the training protocol but may attempt privacy attacks.

3.3. Local Training with Feature Splitting and Model-Data Constraint

Since multi-view data are stored across distributed clients, we analyze the case of client m as an example. Each client constructs a network structure $f_{\theta^m}^m(\cdot)$ that can handle specific views based on its local data, where θ^m are learnable parameters. These networks aim to project the raw data into a low-dimensional space and capture informative latent features. For the client with data of the m-th view type $\mathbf{X}^m$, the latent features can be obtained by $\mathbf{Z}^m = f_{\theta^m}^m(\mathbf{X}^m)$. Based on the latent features, we aim to split them into clustering-related features and sample-related features, defined as $\mathbf{Z}_c^m \in \mathbb{R}^{N_m \times d_m}$ and $\mathbf{Z}_x^m \in \mathbb{R}^{N_m \times d_m}$ respectively.

We hope that the clustering-related features are primarily used to identify the similarities and differences among samples, and are highly correlated with the final clustering results $\mathbf{Y}$. In contrast, the sample-related features are mainly used to describe the properties and details of each individual sample. Based on this idea, we construct the following optimization objective during local training on each client:

$$\mathcal{L}_{\mathbf{X}^m}(w^m) = -I(\mathbf{Z}_x^m; \mathbf{X}^m) - I(\mathbf{Z}_c^m; \mathbf{Y}) + I(\mathbf{Z}_x^m; \mathbf{Z}_c^m). \tag{2}$$

Combining the three mutual information terms achieves local training with feature splitting. The term $-I(\mathbf{Z}_x^m; \mathbf{X}^m)$ represents maximizing the mutual information between the sample-related features $\mathbf{Z}_x^m$ and the raw data $\mathbf{X}^m$, which helps $\mathbf{Z}_x^m$ capture as much information about $\mathbf{X}^m$ as possible, thereby better describing the attributes and details of each sample. Similarly, the term $-I(\mathbf{Z}_c^m; \mathbf{Y})$ represents

maximizing the mutual information between the clustering-related features $\mathbf{Z}_c^m$ and the global clustering results $\mathbf{Y}$, which helps $\mathbf{Z}_c^m$ better reflect the similarities and differences among samples, thereby mining local clustering structures. It is noted that the global clustering results $\mathbf{Y}$ are obtained by the server integrating the shared information from all clients and are optimized after each communication round. Additionally, the term $I(\mathbf{Z}_x^m; \mathbf{Z}_c^m)$ represents minimizing the mutual information between the clustering-related features $\mathbf{Z}_c^m$ and the sample-related features $\mathbf{Z}_x^m$, which aims to reduce the redundant information between $\mathbf{Z}_c^m$ and $\mathbf{Z}_x^m$, achieving high-quality feature splitting. Further, we obtain $\mathbf{Z}_c^m$ and $\mathbf{Z}_x^m$ from latent features by designing two adaptive feature projection layers $\phi_c^m(\cdot)$ and $\phi_x^m(\cdot)$:

$$\mathbf{Z}_c^m = \phi_c^m(\mathbf{Z}^m),\ \mathbf{Z}_x^m = \phi_x^m(\mathbf{Z}^m). \tag{3}$$

To make the above feature splitting easy to handle, by constructing the reconstruction network $g_{w_r^m}(\cdot)$ and the clustering layer $g_{w_c^m}(\cdot)$ separately, with w_r^m and w_c^m as learnable parameters, we derive an objective equal to the original objective in Eq. (2) for client m (see Appendix C.2 for more details) as follows:

$$\begin{aligned}\mathcal{L}_{\mathbf{X}^m}(w^m) =& \frac{1}{N_m}\sum_{i=1}^{N_m}\|\mathbf{x}_i^m - g_{w_r}\left(\mathbf{z}_{i,x}^m\right)\|^2 \\ &+ \frac{1}{N_m}\sum_{i=1}^{N_m}\sum_{k=1}^{K}\mathbf{y}_{i,k}\log\left(g_{w_c}\left(\mathbf{z}_{i,c}^m\right)\right) \\ &+ \frac{1}{N_m}\sum_{i=1}^{N_m}\log p\left(\mathbf{z}_{i,x}^m, \mathbf{z}_{i,c}^m\right).\end{aligned} \tag{4}$$

The underlying insight of the objective function in Eq. (4) is intuitive. Specifically, feature splitting decomposes the latent features $\mathbf{Z}^m$ into two functionally distinct and mutually exclusive types of features. For clarity in the subsequent analysis, we define the three loss components that constitute $\mathcal{L}_{\mathbf{X}^m}(w^m)$ as $\mathcal{L}_1^m$, $\mathcal{L}_2^m$, and $\mathcal{L}_3^m$. The first loss component $\mathcal{L}_1^m$, aims for the sample-related features $\mathbf{Z}_x^m$ to accurately reconstruct the raw data $\mathbf{X}^m$. The second loss component $\mathcal{L}_2^m$, seeks to ensure that the clustering-related features $\mathbf{Z}_c^m$ correctly identify clustering structures. The third loss component $\mathcal{L}_3^m$, imposes a constraint to minimize the correlation between $\mathbf{Z}_x^m$ and $\mathbf{Z}_c^m$. Notably, computing $p\left(\mathbf{z}_{i,x}^m, \mathbf{z}_{i,c}^m\right)$ directly is complicated. So we instead estimate it using histograms, discretizing the continuous variables $\mathbf{z}_{i,x}^m$ and $\mathbf{z}_{i,c}^m$ into bins and approximating the joint distribution based on frequency counts (Scott, 2015).

Building on the above loss terms, we introduce model-data mutual information constraint $I(w^m; \mathbf{X}^m)$ as a regularization term. This constraint limits the complexity of the model's search space, encouraging local models to capture fundamental information and enhancing the generalization ability of the method. The original optimization problem in Eq. (4) is thus reformulated as:

$$\min_{p(w^m|\mathbf{X}^m)} \mathcal{L}^m = \mathbb{E}_{p(w^m|\mathbf{X}^m)}\left[\mathcal{L}_{\mathbf{X}^m}(w^m) + \alpha I\left(w^m; \mathbf{X}^m\right)\right], \tag{5}$$

where $\alpha > 0$ is a parameter that balances fitting and generalization. Furthermore, inspired by previous works (Wang et al., 2021; Zhang et al., 2024a), we adopt Bayesian inference to transform the single-point estimation of the local model w^m into a distributional estimation $p(w^m \mid \mathbf{X}^m)$, facilitating subsequent optimization. The following lemma is provided to find an optimal posterior.

Lemma 3.1. *(Wang et al., 2021) Given an observed dataset $\mathbf{X}^m$, the optimal posterior $p(w^m \mid \mathbf{X}^m)$ for client m update in Eq. (5) should satisfy the following form that*

$$p\left(w^m \mid \mathbf{X}^m\right) = \frac{1}{B}\exp\left\{-\frac{1}{\alpha}U(w^m)\right\}, \tag{6}$$

where B is a normalizing factor that makes the integral of the distribution equal to 1, and $U(w^m)$ is the energy function defined as $U(w^m) = \mathcal{L}_{\mathbf{X}^m}(w^m) - \alpha \log p(w^m)$.

Please see Appendix C.3 for the proof. The resulting optimal posterior follows a typical Gibbs distribution (Kittel, 2004) with energy function $U(w^m)$ and temperature α (the same α appears in Eq. (5)). Therefore, we use stochastic gradient Langevin dynamics (SGLD) (Welling & Teh, 2011) to optimize and obtain this Gibbs posterior, which has been shown to be effective and scalable for large-scale posterior inference problems. Specifically, SGLD can be implemented through a straightforward adaptation of stochastic gradient descent, as follows

$$w_{t,k}^m = w_{t,k-1}^m - \eta_{t,k}\nabla U\left(w_{t,k-1}^m\right) + \sqrt{2\eta_{t,k}\alpha}\zeta_{t,k}, \tag{7}$$

where $w_{t,k}^m$ represents client m's model at communication round t and step k, $\eta_{t,k}$ is the local step size, $\nabla U\left(w_{t,k-1}^m\right)$ is an unbiased estimate of energy function gradient, and $\zeta_{t,k} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is a standard Gaussian noise vector.

After clients complete the current round of local training, they retain the sample-related features locally to prevent privacy leakage while sharing the clustering-related features with the server to mine complementary clustering structures.

3.4. Global Training with Shared Information

In our setting, the server does not require any additional information, such as public datasets or pre-trained models. It only uses the limited information shared by clients, i.e., clustering-related features, to mine high-quality global clustering structures. Additionally, for semi-honest participants, namely, participants faithfully execute the training protocol

but may launch privacy attacks to infer other parties' private data, having only clustering-related features also prevents them from reconstructing the raw data by existing attack methods, as shown in Figure 3 (c).

After receiving the clustering-related features uploaded by each client, the server concatenates them to generate the global features:

$$\mathbf{Z} = \left[\mathbf{Z}^1, \mathbf{Z}^2, ..., \mathbf{Z}^M\right] \in \mathbb{R}^{N \times \sum_{m=1}^{M} d_m}. \tag{8}$$

For the clustering task, the server applies K-means (MacQueen, 1967) on the global features $\mathbf{Z}$ obtained from Eq. (6) to mine complementary global clustering structures. Letting $\{\mathbf{c}_j\}_{j=1}^{K}$ denote the K cluster centroids, we have:

$$\min_{\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_K} \sum_{i=1}^{N} \sum_{j=1}^{K} \left\|\mathbf{z}_i - \mathbf{c}_j\right\|^2. \tag{9}$$

Furthermore, the global clustering results are calculated by

$$y_i = \arg\min_j \left\|\mathbf{z}_i - \mathbf{c}_j\right\|^2 \in \mathbf{Y}, \tag{10}$$

where $\mathbf{Y}$ represents the global clustering results, which are distributed to all clients to aid in performing high-quality feature splitting and sharing clustering-related features.

3.5. Extending to Incomplete Scenarios

Due to the inherent flexibility of ESFMC, it can be easily extended to handle incomplete scenarios, where the samples in all clients only partially overlap, meaning the number of client samples N_m is not equal to the global sample number N. To maintain clustering performance in more complex scenarios, inevitably, more shared information exposure is required. Therefore, to protect privacy while dealing with incomplete scenarios, we introduce shared information clustering assignments $\mathbf{Q}^m$. These are extracted locally by each client and then uploaded to the server, which aggregates them to obtain global clustering assignments $\mathbf{Q}$ and distributes them to all clients for collaborative training.

For client m, the clustering-related features $\mathbf{Z}_c^m$ are processed through the clustering layer $g_{w_c}(\cdot)$ to produce clustering assignments $\mathbf{Q}^m \in \mathbb{R}^{N_m \times K}$, where $\mathbf{Q}^m = g_{w_c}(\mathbf{Z}_c^m)$. These clustering assignments are then uploaded to the server. During local training, we introduce an additional loss to explore non-overlapping samples among clients, thus better addressing incomplete scenarios. Consequently, we design a collaborative alignment strategy to maximize mutual information $I(\mathbf{Z}_c^m; \tilde{g}_{w_z}(\mathbf{Q}))$, ensuring that the information shared among different clients remains consistent from the global perspective, thereby reducing errors due to incomplete scenarios. Here, $\tilde{g}_{w_z}(\cdot) : \mathbb{R}^K \to \mathbb{R}^{d_m}$ aims to map the global clustering assignments to the latent feature space corresponding to the view type data. Similarly, we express the optimization objective in a more intuitive form:

$$\begin{aligned}
\mathcal{L}_a^m =& -I(\mathbf{Z}_c^m; \tilde{g}_{w_z}(\mathbf{Q})) \\
=& H\left(\mathbf{Z}_c^m \mid \tilde{g}_{w_z}(\mathbf{Q})\right) - H\left(\mathbf{Z}_c^m\right) \\
=& -\mathbb{E}_{p(\mathbf{z}_c^m, \tilde{g}_{w_z}(\mathbf{q}))}\left[\log p\left(\mathbf{z}_c^m \mid \tilde{g}_{w_z}(\mathbf{q})\right)\right] \\
&+ \mathbb{E}_{p(\mathbf{z}_c^m)}\left[\log p\left(\mathbf{z}_c^m\right)\right] \\
\leqslant& \frac{1}{N_m} \sum_{i=1}^{N_m} \left\|\mathbf{z}_{i,c}^m - \tilde{g}_{w_z}\left(\mathbf{q}_i\right)\right\|^2.
\end{aligned} \tag{11}$$

Since $p\left(\mathbf{z}_c^m\right)$ is difficult to estimate, we directly optimize the upper bound of $\mathcal{L}_a^m$. Thus, in the incomplete scenario, the total loss for client m is represented as:

$$\mathcal{L}_{\text{inc}}^m = \mathcal{L}^m + \mathcal{L}_a^m. \tag{12}$$

After clients complete the current round of local training, they upload the clustering-related features $\tilde{\mathbf{Z}}_c^m$ optimized from the global perspective by Eq. (13) and local clustering assignments $\mathbf{Q}^m$ to the server.

$$\tilde{\mathbf{Z}}_c^m = [\mathbf{Z}_c^m; \tilde{g}_{w_z}\left(\mathbf{Q}^g\right)] \in \mathbb{R}^{N \times d_m}, \tag{13}$$

where we define $\mathbf{Q}^g = \{\mathbf{q}_i \mid \mathbf{q}_i \in \mathbf{Q} \Lambda \mathbf{q}_i \notin \mathbf{Q}^m\}$.

The server receives clustering-related features $\tilde{\mathbf{Z}}_c^m$ and local clustering assignments $\mathbf{Q}^m$ uploaded by each client. Then, it can calculate clustering results $\mathbf{Y}$ by Eqs. (8)-(10) and acquire global cluster assignments $\mathbf{Q}$ by following formula:

$$\mathbf{q}_i = \frac{\sum_{m=1}^{M} \mathbf{I}_{im} \mathbf{q}_i^m}{\sum_{m=1}^{M} \mathbf{I}_{im}} \in \mathbf{Q}, \tag{14}$$

where $\mathbf{I} \in \{0, 1\}^{N \times M}$ is the indicator matrix, $\mathbf{I}_{im} = 1$ if the i-th sample exists in the m-th client; otherwise, $\mathbf{I}_{im} = 0$. In incomplete scenarios, we extend the general ESFMC method to further explore how to maintain a balance between privacy protection and performance improvement.

In incomplete scenarios, we can opt not to share additional information, such as clustering assignments, and instead rely on the model's generalization performance. Specifically, each client uploads clustering-related features for overlapping samples to the server, while non-overlapping samples are clustered locally. This strategy is particularly effective when samples across clients exhibit high overlap, as shown in Figure 3 (b).

4. Discussion and Analysis

4.1. Generalization Analysis

We denote the population risk as $L_{\mathcal{P}}(w)$ and recall the global empirical risk $L_{\mathcal{X}}(w)$ in Eq. (5), then the expected generalization error can be denoted as $\mathbb{E}\left[L_{\mathcal{P}}(w) - L_{\mathcal{X}}(w)\right]$.

Theorem 4.1. *Suppose that $\ell_m\left(w^m, \mathbf{x}_i^m\right)$ for all $m \in M$ is bounded by C and independent, then the expected generalization error satisfies*

$$\mathbb{E}\left[L_{\mathcal{P}}(w) - L_{\mathcal{X}}(w)\right] \leq \frac{C}{M} \sum_{m=1}^{M} \sqrt{\frac{I(w^m; \mathbf{X}^m)}{2N_m}}, \quad (15)$$

where M is the number of participating clients and N_m is the number of samples in the m-th participating client.

Please see Appendix C.4 for the detailed proof. The above theorem shows that increasing the number of participating clients or the sample size per client improves generalization performance. Furthermore, the mutual information constraint $I(w^m; \mathbf{X}^m)$ introduced during local training is strongly correlated with generalization error, theoretically validating the effectiveness of our method.

We can rely on the model's generalization ability to extend it to incomplete and cross-device scenarios. Specifically, for incomplete scenarios, each client uploads clustering-related features for overlapping samples to the server, while non-overlapping samples are clustered locally. For cross-device scenarios, each client uploads clustering-related features extracted from local data, with the server aligning overlapping samples. Our method leverages generalization to perform well, as shown in Figure 3 (b) and Figure 4.

4.2. Privacy Analysis

We divide the privacy analysis into two parts: differential privacy for each client and the overall privacy analysis for ESFMC. For each client, our approach ensures client-level (ε, δ)-differential privacy through posterior inference based on SGLD sampling and feature splitting, as detailed in Appendix C.5. Notably, our analysis proves that sharing the clustering-related features $\mathbf{Z}_c^m$ provides stronger privacy protection than directly sharing the latent features $\mathbf{Z}^m$. Then, by combining the privacy analysis for each client with the composition theorem (Kairouz et al., 2015), we obtain the overall privacy guarantee.

Theorem 4.2. *For M clients with (ε, δ)-differential privacy, ESFMC satisfies $\left(\tilde{\varepsilon}_{\tilde{\delta}}, 1-(1-\delta)^M(1-\tilde{\delta})\right)$-differential privacy, where*

$$\tilde{\varepsilon}_{\tilde{\delta}} = \min\Bigg\{M\varepsilon, \frac{\left(e^{\varepsilon}-1\right)\varepsilon M}{e^{\varepsilon}+1} + \varepsilon\sqrt{2M\log\left(e + \frac{\sqrt{M\varepsilon^2}}{\tilde{\delta}}\right)}, \frac{\left(e^{\varepsilon}-1\right)\varepsilon M}{e^{\varepsilon}+1} + \varepsilon\sqrt{2M\log\left(\frac{1}{\tilde{\delta}}\right)}\Bigg\},$$

and $\varepsilon = (\sqrt{2\log(1.25/\delta)}/\sigma + \rho_c\sqrt{k\log(1/\delta)}/\sigma_c)$.

Our method is primarily designed for environments where all participating parties are semi-honest, meaning they faithfully execute the training protocol but may attempt privacy attacks. Currently, our feature splitting strategy is sufficient to defend against common model inversion attacks in such settings, as shown in Figure 3 (c). For further privacy enhancement, we could integrate commonly used privacy-preserving techniques in federated learning, such as homomorphic encryption (Cheng et al., 2021) or secure multi-party computation (Gu et al., 2021), to offer additional privacy protection.

4.3. Complexity Analysis

Suppose K, M, and N represent the number of clusters, clients, and total samples, respectively. Let H denote the maximum number of hidden neurons in the clients' networks. And Z denotes the maximum dimensionality of latent features. Generally $N \gg V, K, M$ holds. For client m, the complexity is $O(NH + NKH + NZ^2)$. For the server, the complexity is $O(NMZK)$. In conclusion, due to M clients running in parallel, the total complexity of our algorithm is $O(NKH + NZ^2 + NMZK)$ in each iteration, which is linear to the data size N.

5. Experiment

5.1. Experimental Settings

Datasets. We conduct the experiments on six public multi-view datasets. Specifically, **Caltech** (Fei-Fei et al., 2004) contains 1400 RGB images in 7 categories, and includes five views. **HW**[1] consists of 10 categories, each corresponding to the digits from 0 through 9, and includes a total of 2,000 samples, with each sample represented by six views. **MNIST-USPS** (Peng et al., 2019) comprises 5000 samples collected from two handwritten digital image datasets, which are considered as two views. **Synthetic3d** (Kumar et al., 2011) comprises 3 categories, containing a total of 600 samples, each with three views. **BDGP** (Cai et al., 2012) encompasses 2,500 samples representing 5 different types of Drosophila embryos. **Scene** (Fei-Fei & Perona, 2005) contains 4,485 scene images, each captured from three views, across 15 categories. Our code is available at https://github.com/5Martina5/ESFMC.

Note that in our federated setting, the multiple views of these datasets are distributed across different clients and are isolated from each other. Additionally, to evaluate the effectiveness of our method in incomplete scenarios, we randomly remove some samples from arbitrary views. Furthermore, we introduce the sample overlapping rate $\phi = m/n$ following (Chen et al., 2023), where n represents the total size of the dataset and m denotes the number of samples that have fully overlapping views across all clients.

[1] https://archive.ics.uci.edu/ml/datasets.php

Table 1. Experiments on four datasets in general scenarios. The best result is shown in **bold** and the second-best is underlined.

Methods	Caltech			HW			MNIST-USPS			Synthetic3d		
	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
GIMC-FLSD	48.60	35.12	28.25	42.15	47.36	29.80	79.71	76.74	68.97	53.39	14.81	9.79
HCP-IMSC	78.37	69.68	64.93	82.55	79.33	74.34	95.82	90.79	91.04	80.00	45.11	49.07
IMVC-CBG	37.61	31.43	18.96	60.38	61.79	47.95	53.35	48.77	35.66	45.83	12.32	10.73
DSIMVC	78.86	67.50	60.53	81.70	79.23	73.50	**98.08**	**94.81**	**95.05**	94.83	81.10	85.31
AGDIMC	90.14	82.91	80.27	97.35	93.53	93.73	91.60	85.72	84.21	59.67	56.89	45.98
FedDMVC	89.07	81.11	78.52	96.48	92.52	92.35	84.56	89.73	82.29	58.24	56.28	43.58
FedMVFCM	59.37	55.40	53.47	64.14	64.57	56.22	58.12	53.05	47.60	93.07	75.78	87.03
FedMVFPC	43.38	31.30	22.78	52.67	40.85	33.10	48.48	41.90	29.05	91.74	75.55	77.59
FCUIF	89.14	80.79	78.24	97.85	95.16	95.30	82.92	83.06	74.13	63.67	58.57	48.27
ESFMC (ours)	**91.50**	**84.54**	**83.45**	**98.02**	**95.34**	**95.57**	97.24	93.19	94.02	**97.32**	**87.67**	**91.23**

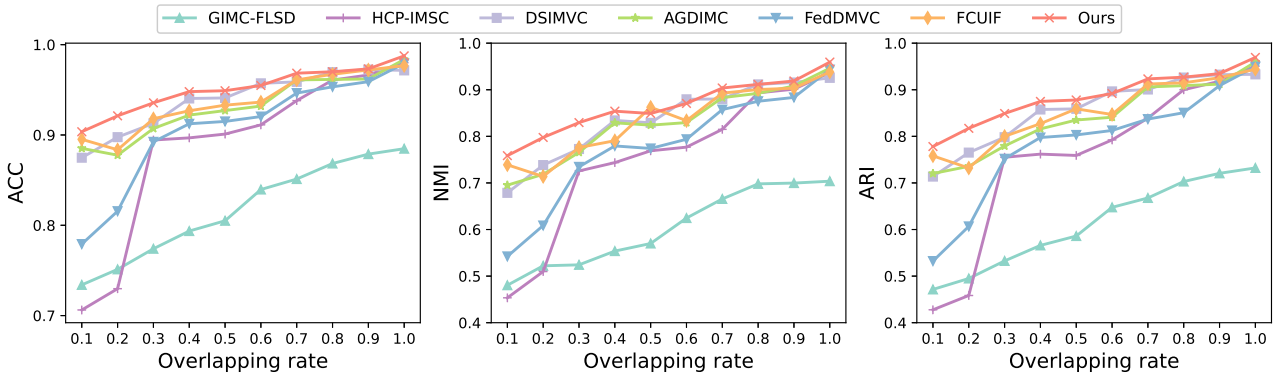


Figure 2. Performance analysis on BDGP with different overlapping rates.

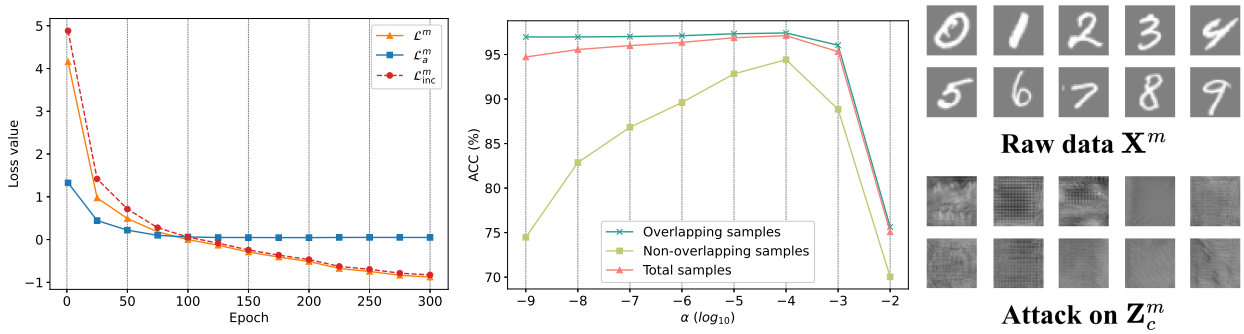
Table 2. Clustering results in incomplete scenarios. The best result is shown in **bold** and the second-best is underlined.

Methods	BDGP			Scene		
	ACC	NMI	ARI	ACC	NMI	ARI
GIMC-FLSD	80.5	57.0	58.6	30.0	26.4	13.5
HCP-IMSC	90.1	76.9	75.9	32.5	27.3	14.3
IMVC-CBG	36.3	17.6	5.6	26.8	27.0	14.4
DSIMVC	94.1	82.9	85.9	27.8	30.4	14.5
AGDIMC	92.7	82.4	83.5	38.6	32.7	20.3
FedDMVC	91.5	77.4	80.3	39.3	34.3	22.5
FCUIF	93.3	**86.2**	85.9	**41.0**	37.6	22.9
Ours	**94.9**	84.9	**87.8**	40.8	**42.4**	**26.2**

Comparing Methods. To demonstrate the performance of our proposed ESFMC, we select several relevant algorithms as comparison methods. These include five centralized multi-view clustering methods, i.e., GIMC-FLSD (Wen et al., 2020), HCP-IMSC (Li et al., 2022), IMVC-CBG (Wang et al., 2022), DSIMVC (Tang & Liu, 2022) and AGDIMC (Pu et al., 2024), and four federated multi-view clustering methods, i.e., FedDMVC (Chen et al., 2023), FedMVFCM (Hu et al., 2023), FedMVFPC (Hu et al., 2023) and FCUIF (Ren et al., 2024). Notably, FedMVFCM and FedMVFPC are not applicable to incomplete scenarios, so they are excluded from comparison in this scenario.

5.2. Clustering Results

Tables 1 and 2 present the quantitative comparisons of ESFMC with baselines under both general scenarios and incomplete scenarios. Each experiment is independently repeated five times, and the average values are reported. In Table 1, it can be observed that the proposed method achieves the best performance across the Caltech, MNIST-USPS, and Synthetic3d datasets, with particularly significant improvements on the Synthetic3d dataset compared to the second-best method. These results demonstrate that ESFMC effectively maintains strong clustering performance while ensuring privacy, thereby alleviating the trade-off between privacy protection and performance enhancement. The success of ESFMC encourages us to focus on methodological



(a) Loss $vs.$ Epoch. (b) ACC $vs.$ α. (c) Attack results on ESFMC.

Figure 3. (a) Convergence analysis on BDGP. (b) Parameter analysis on BDGP. (c) Privacy verification on MNIST-USPS.

improvements directly related to the task. It enhances performance and further reduces information redundancy.

Table 2 reports the results of ESFMC and the comparison methods under incomplete scenarios with the overlapping rate of $\phi = 0.5$. The findings indicate that the extended ESFMC performs well in handling incomplete scenarios with limited information exposure, achieving strong clustering results on both the BDGP and Scene datasets. Additionally, we conduct experiments on the BDGP dataset, exploring varying overlapping rates from 0.1 to 1 with an interval of 0.1, as shown in Figure 2. The figure for IMVC-CBG is omitted due to poor performance, to highlight clearer variations. The results show that ESFMC adapts to different levels of client sample overlap, with clustering performance improving as the overlapping rate increases.

5.3. Model Analysis

Ablation Study. To further validate the contributions of the proposed method, we conduct ablation studies, as shown in Table 3. "w/o" indicates the absence of a specific loss component in the method. The results show that $\mathcal{L}_1^m$ and $\mathcal{L}_3^m$ have minimal impact on clustering performance. Their primary role is to ensure that the shared features $\mathbf{Z}_c^m$, derived after feature splitting in the client, contain only clustering-related information, while sample-related information remains local to the client, thus preventing privacy leakage. In contrast, $\mathcal{L}_2^m$ significantly influences the global clustering performance as it involves the extraction of the shared information $\mathbf{Z}_c^m$. Additionally, in incomplete scenarios, $\mathcal{L}_a^m$ also impacts global clustering performance, demonstrating the effectiveness of the proposed strategy.

Convergence Analysis. We conduct the convergence analysis of our method in incomplete scenarios on the BDGP dataset with the overlapping rate of $\phi = 0.5$. Figure 3 (a) depicts the changes in $\mathcal{L}^m$, $\mathcal{L}_a^m$, and the total loss $\mathcal{L}_{\text{inc}}^m$ over the epochs, with the results reported for a randomly selected client. The results demonstrate that all loss components gradually converge and stabilize as the number of epochs increases, indicating that ESFMC is stable and effective.

Table 3. Ablation study on two datasets with different settings. The best in each column is shown in **bold**.

Settings	Variants	ACC	NMI	ARI
In general scenarios (Caltech)	w/o $\mathcal{L}_1^m$	79.78	73.16	66.64
	w/o $\mathcal{L}_2^m$	26.78	7.74	4.86
	w/o $\mathcal{L}_3^m$	87.28	78.74	76.22
	ESFMC	**91.50**	**84.54**	**83.45**
In incomplete scenarios (BDGP)	w/o $\mathcal{L}_1^m$	94.48	84.29	86.78
	w/o $\mathcal{L}_2^m$	25.04	2.55	2.15
	w/o $\mathcal{L}_3^m$	94.44	85.16	87.3
	w/o $\mathcal{L}_a^m$	86.24	68.29	68.97
	ESFMC	**94.93**	**84.89**	**87.82**

Parameter Analysis. Reviewing Eq. (5), α serves as a parameter to adjust $I(w^m; \mathbf{X}^m)$, balancing fitting and generalization. Figure 3 (b) illustrates the clustering performance for different samples on the BDGP dataset with an overlapping rate of $\phi = 0.9$ under varying α. Here we adopt the strategy of clustering non-overlapping samples depending on the model's generalization performance, implying a positive correlation between generalization and clustering performance. As α increases, clustering performance for overlapping samples initially shows little change, but drops sharply when $\alpha = 10^{-2}$, indicating that excessive simplification of the model harms clustering performance. Additionally, as α increases, the clustering performance for non-overlapping and total samples first increases, then decreases. This shows that when α is small, the model overfits the training data, weakening its generalization and negatively impacting overall clustering performance. When $\alpha = 10^{-4}$, the model balances fitting and generalization. Additionally, when α is small, it effectively removes the constraint term $I(w^m; \mathbf{X}^m)$, which can be viewed as an

ablation analysis of the constraint.

Privacy Verification. We provide empirical analysis to support ESFMC's privacy guarantee. Specifically, we assess whether all semi-honest participants can reconstruct the original data through certain attack methods based on shared information. To simulate potential attacks, we apply the widely-used model inversion attack (He et al., 2019) to reconstruct data from the MNIST-USPS dataset using the shared clustering-related features, as shown in Figure 3 (c). The results indicate that while achieving good performance, ESFMC also enhances privacy protection.

6. Conclusion

In this paper, we propose ESFMC, which is designed to alleviate the trade-off between privacy preservation and performance improvement in FedMVC. Specifically, we split the features extracted locally by clients to reduce privacy risks while ensuring the server can mine complementary global clustering structures. Additionally, we extend the proposed method to incomplete scenarios by designing a collaborative alignment strategy. We further analyze the generalization performance and privacy guarantees of our method, confirming its effectiveness and security. Experimental results on multiple real-world multi-view datasets demonstrate that our method outperforms state-of-the-art methods in effectiveness and security.

Acknowledgements

This work was supported in part by National Key Research and Development Program of China (Nos. 2024YFC2310800 and 2024YFC2310801), National Natural Science Foundation of China (No. 62476052), Sichuan Science and Technology Program (No. 2024NSFSC1473), and Central Guidance for Local Science and Technology Development Fund Projects (No. 2024ZYD0268).

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., and Zhang, L. Deep learning with differential privacy. In *ACM CCS*, pp. 308–318, 2016.

Alquier, P. et al. User-friendly introduction to pac-bayes bounds. *Foundations and Trends® in Machine Learning*, 17:174–303, 2024.

Cai, X., Wang, H., Huang, H., and Ding, C. Joint stage recognition and anatomical annotation of drosophila gene expression patterns. *Bioinformatics*, 28(12):i16–i24, 2012.

Chao, G., Wang, S., Yang, S., Li, C., and Chu, D. Incomplete multi-view clustering with multiple imputation and ensemble clustering. *Applied Intelligence*, 52(13): 14811–14821, 2022.

Chao, G., Jiang, Y., and Chu, D. Incomplete contrastive multi-view clustering with high-confidence guiding. In *AAAI*, volume 38, pp. 11221–11229, 2024.

Che, S., Kong, Z., Peng, H., Sun, L., Leow, A., Chen, Y., and He, L. Federated multi-view learning for private medical data integration and analysis. *TIST*, 13(4):1–23, 2022.

Chen, X., Xu, J., Ren, Y., Pu, X., Zhu, C., Zhu, X., Hao, Z., and He, L. Federated deep multi-view clustering with global self-supervision. In *ACM MM*, pp. 3498–3506, 2023.

Chen, X., Ren, Y., Xu, J., Lin, F., Pu, X., and Yang, Y. Bridging gaps: Federated multi-view clustering in heterogeneous hybrid views. *NeurIPS*, 37:37020–37049, 2024.

Cheng, K., Fan, T., Jin, Y., Liu, Y., Chen, T., Papadopoulos, D., and Yang, Q. Secureboost: A lossless federated learning framework. *IEEE intelligent systems*, 36(6):87–98, 2021.

Cruickshank, I. *Multi-view Clustering of Social-based Data.* PhD thesis, Carnegie Mellon University, USA, 2020.

Dai, H., Liu, Y., Su, P., Cai, H., Huang, S., and Lv, J. Multi-view clustering by inter-cluster connectivity guided reward. In *ICML*, pp. 1–10, 2024.

Duan, L. L. Latent simplex position model: High dimensional multi-view clustering with uncertainty quantification. *JMLR*, 21(38):1–25, 2020.

Dwork, C., Roth, A., et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9:211–407, 2014.

Fang, U., Li, M., Li, J., Gao, L., Jia, T., and Zhang, Y. A comprehensive survey on multi-view clustering. *TKDE*, 35:12350–12368, 2023.

Fei-Fei, L. and Perona, P. A bayesian hierarchical model for learning natural scene categories. In *CVPR*, pp. 524–531, 2005.

Fei-Fei, L., Fergus, R., and Perona, P. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *CVPR*, pp. 59–70, 2004.

Flanagan, A., Oyomno, W., Grigorievskiy, A., Tan, K. E., Khan, S. A., and Ammad-Ud-Din, M. Federated multi-view matrix factorization for personalized recommendations. In *ECML-PKDD*, pp. 324–347, 2021.

Geiping, J., Bauermeister, H., Dröge, H., and Moeller, M. Inverting gradients-how easy is it to break privacy in federated learning? In *NeurIPS*, pp. 16937–16947, 2020.

Glorot, X., Bordes, A., and Bengio, Y. Deep sparse rectifier neural networks. In *AISTATS*, pp. 315–323, 2011.

Gu, B., Xu, A., Huo, Z., Deng, C., and Huang, H. Privacy-preserving asynchronous vertical federated learning algorithms for multiparty collaborative learning. *TNNLS*, 33 (11):6103–6115, 2021.

He, Z., Zhang, T., and Lee, R. B. Model inversion attacks against collaborative inference. In *ACSAC*, pp. 148–162, 2019.

Hellström, F., Durisi, G., Guedj, B., and Raginsky, M. Generalization bounds: Perspectives from information theory and pac-bayes. *arXiv preprint arXiv:2309.04381*, 2023.

Hu, S., Yan, X., and Ye, Y. Dynamic auto-weighted multi-view co-clustering. *Pattern Recognition*, 99:1–12, 2020.

Hu, X., Qin, J., Shen, Y., Pedrycz, W., Liu, X., and Liu, J. An efficient federated multi-view fuzzy c-means clustering method. *TFS*, 2023.

Huang, H., Zhou, G., Zheng, Y., Qiu, Y., Wang, A., and Zhao, Q. Adversarially robust deep multi-view clustering: A novel attack and defense framework. In *ICML*, pp. 1–33, 2024.

Huang, S., Shi, W., Xu, Z., Tsang, I. W., and Lv, J. Efficient federated multi-view learning. *Pattern Recognition*, 131: 1–10, 2022.

Huang, Y., Guo, K., Yi, X., Li, Z., and Li, T. Incremental unsupervised feature selection for dynamic incomplete multi-view data. *Information Fusion*, 96:312–327, 2023.

Jiang, X., Ma, Z., Fu, Y., Liao, Y., and Zhou, P. Heterogeneity-aware federated deep multi-view clustering towards diverse feature representations. In *ACM MM*, pp. 1–9, 2024.

Kairouz, P., Oh, S., and Viswanath, P. The composition theorem for differential privacy. In *ICML*, pp. 1376–1385, 2015.

Kittel, C. *Elementary statistical physics*. Courier Corporation, 2004.

Kumar, A., Rai, P., and Daume, H. Co-regularized multi-view spectral clustering. *NeurIPS*, 24, 2011.

Lee, M. and Pavlovic, V. Private-shared disentangled multimodal vae for learning of latent representations. In *ICCV*, pp. 1692–1700, 2021.

Li, Z., Tang, C., Zheng, X., Liu, X., Zhang, W., and Zhu, E. High-order correlation preserved incomplete multi-view subspace clustering. *TIP*, 31:2067–2080, 2022.

Lin, J., Zheng, Y., Chen, X., Ren, Y., Pu, X., and He, J. Cross-view contrastive unification guides generative pre-training for molecular property prediction. In *ACM MM*, pp. 1–9, 2024.

Lin, Y.-M., Gao, Y., Gong, M.-G., Zhang, S.-J., Zhang, Y.-Q., and Li, Z.-Y. Federated learning on multimodal data: A comprehensive survey. *MIR*, 20(4):539–553, 2023.

Liu, Z., Huang, H., Letchmunan, S., and Deveci, M. Adaptive weighted multi-view evidential clustering with feature preference. *KBS*, 294:1–18, 2024.

Lu, S., Xu, D., Zhang, C., and Zhu, Z. Fast dynamic multi-view clustering with semantic-consistency inheritance. *KBS*, pp. 1–10, 2024a.

Lu, Y., Lin, Y., Yang, M., Peng, D., Hu, P., and Peng, X. Decoupled contrastive multi-view clustering with high-order random walks. In *AAAI*, volume 38, pp. 14193–14201, 2024b.

MacQueen, J. Classification and analysis of multivariate observations. In *BSMSP*, pp. 281–297, 1967.

Mahendran, A. and Vedaldi, A. Understanding deep image representations by inverting them. In *CVPR*, pp. 5188–5196, 2015.

Nguyen, N.-B., Chandrasegaran, K., Abdollahzadeh, M., and Cheung, N.-M. Re-thinking model inversion attacks against deep neural networks. In *CVPR*, pp. 16384–16393, 2023.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019.

Peng, X., Huang, Z., Lv, J., Zhu, H., and Zhou, J. T. Comic: Multi-view clustering without parameter selection. In *ICML*, pp. 5092–5101, 2019.

Pu, J., Cui, C., Chen, X., Ren, Y., Pu, X., Hao, Z., Philip, S. Y., and He, L. Adaptive feature imputation with latent graph for deep incomplete multi-view clustering. In *AAAI*, volume 38, pp. 14633–14641, 2024.

Qiao, D., Ding, C., and Fan, J. Federated spectral clustering via secure similarity reconstruction. In *NeurIPS*, pp. 1–36, 2023.

Ren, Y., Chen, X., Xu, J., Pu, J., Huang, Y., Pu, X., Zhu, C., Zhu, X., Hao, Z., and He, L. A novel federated multi-view clustering method for unaligned and incomplete data fusion. *Information Fusion*, 108:1–10, 2024.

Scott, D. W. *Multivariate density estimation: theory, practice, and visualization*. John Wiley & Sons, 2015.

Sharma, K. K. and Seal, A. Outlier-robust multi-view clustering for uncertain data. *KBS*, 211:1–14, 2021.

Tang, H. and Liu, Y. Deep safe incomplete multi-view clustering: Theorem and algorithm. In *ICML*, pp. 21090–21110, 2022.

Tishby, N., Pereira, F. C., and Bialek, W. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

Wan, X., Xiao, B., Liu, X., Liu, J., Liang, W., and Zhu, E. Fast continual multi-view clustering with incomplete views. *TIP*, 2024.

Wang, H., Yao, M., Chen, Y., Xu, Y., Liu, H., Jia, W., Fu, X., and Wang, Y. Manifold-based incomplete multi-view clustering via bi-consistency guidance. *TMM*, 2024.

Wang, S., Liu, X., Liu, L., Tu, W., Zhu, X., Liu, J., Zhou, S., and Zhu, E. Highly-efficient incomplete large-scale multi-view clustering with consensus bipartite graph. In *CVPR*, pp. 9776–9785, 2022.

Wang, Z., Huang, S.-L., Kuruoglu, E. E., Sun, J., Chen, X., and Zheng, Y. Pac-bayes information bottleneck. In *ICLR*, pp. 1–16, 2021.

Welling, M. and Teh, Y. W. Bayesian learning via stochastic gradient langevin dynamics. In *ICML*, pp. 681–688, 2011.

Wen, J., Zhang, Z., Zhang, Z., Fei, L., and Wang, M. Generalized incomplete multiview clustering with flexible locality structure diffusion. *TCYB*, 51(1):101–114, 2020.

Wen, J., Deng, S., Wong, W., Chao, G., Huang, C., Fei, L., and Xu, Y. Diffusion-based missing-view generation with the application on incomplete multi-view clustering. In *ICML*, pp. 1–17, 2024.

Xu, C., Si, J., Guan, Z., Zhao, W., Wu, Y., and Gao, X. Reliable conflictive multi-view learning. In *AAAI*, volume 38, pp. 16129–16137, 2024.

Yan, X., Wang, Z., and Jin, Y. Federated incomplete multi-view clustering with heterogeneous graph neural networks. *arXiv preprint arXiv:2406.08524*, 2024.

Yang, B., Zhang, X., Nie, F., Wang, F., Yu, W., and Wang, R. Fast multi-view clustering via nonnegative and orthogonal factorization. *TIP*, 30:2575–2586, 2020.

Yang, Z., Zhang, Y., Zheng, Y., Tian, X., Peng, H., Liu, T., and Han, B. Fedfed: Feature distillation against data heterogeneity in federated learning. In *NeurIPS*, pp. 1–32, 2023.

Yin, H., Hu, W., Zhang, Z., Lou, J., and Miao, M. Incremental multi-view spectral clustering with sparse and connected graph learning. *Neural Networks*, 144:260–270, 2021.

Yu, H., Zhang, T., Chen, J., Guo, C., and Lian, Y. Web items recommendation based on multi-view clustering. In *COMPSAC*, volume 1, pp. 420–425, 2018.

Yuan, C., Zhu, Y., Zhong, Z., Zheng, W., and Zhu, X. Robust self-tuning multi-view clustering. *WWW*, 25(2):489–512, 2022.

Zhang, H., Li, C., Kan, N., Zheng, Z., Dai, W., Zou, J., and Xiong, H. Improving generalization in federated learning with model-data mutual information regularization: A posterior inference approach. In *NeurIPS*, pp. 1–33, 2024a.

Zhang, K., Du, S., Wang, Y., and Deng, T. Deep incomplete multi-view clustering via attention-based direct contrastive learning. *Expert Systems with Applications*, pp. 1–14, 2024b.

Zhou, L., Du, G., Lü, K., Wang, L., and Du, J. A survey and an empirical evaluation of multi-view clustering approaches. *ACM Computing Surveys*, 56(7):1–38, 2024.

We provide more details and results about our work in the appendix. Here are the contents:

- Appendix A: Framework of the proposed algorithm.
- Appendix B: More related work.
- Appendix C: Proofs.
- Appendix D: More details about experimental settings.
- Appendix E: Additional experiment results.

A. Framework of the Proposed Algorithm

Algorithm 1 summarizes the pipeline of ESFMC in general scenarios and incomplete scenarios respectively. In general scenarios, local clients perform feature splitting, extracting cluster-related features to be uploaded to the server. The server then utilizes this limited shared information to mine global clustering structures and distributes them back to the clients to assist them in feature splitting. In incomplete scenarios, in addition to the steps performed in general scenarios, alignment loss is introduced to ensure alignment among non-overlapping samples across clients. This requires sharing clustering assignments to facilitate training.

Algorithm 1 The optimization of ESFMC

input Data with M views $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, ..., \mathbf{X}^M\}$, which are distributed on M clients, number of clusters K, communication round R.
output Global clustering predictions $\mathbf{Y}$.

(In General Scenarios)
while not reaching R rounds **do**
  for $m = 1$ to M in parallel **do**
    Perform feature splitting by Eqs. (3, 5).
    Use SGLD to optimize the local model by Eq. (7).
    Upload cluster-related features $\mathbf{Z}_c^m$ to the server.
  end for
  Update global features $\mathbf{Z}$ by Eq. (8).
  Calculate clustering predictions $\mathbf{Y}$ by Eqs. (9)-(10).
  Distribute $\mathbf{Y}$ to each client.
end while
(In Incomplete Scenarios)
while not reaching R rounds **do**
  for $m = 1$ to M in parallel **do**
    Optimize the total loss by Eqs. (5, 7, 11, 12).
    Upload cluster-related features $\mathbf{Z}_c^m$ and local clustering assignments $\mathbf{Q}^m$ to the server.
  end for
  Calculate clustering predictions $\mathbf{Y}$ by Eqs. (8)-(10).
  Obtain global clustering assignments $\mathbf{Q}$ by Eq. (14).
  Distribute $\mathbf{Y}$ and $\mathbf{Q}$ to each client.
end while

B. More Related Work

The diversity and large scale of multi-view data present greater challenges to traditional clustering techniques. Unlike single-view clustering, Multi-view clustering (MVC) can integrate information from different perspectives to reveal hidden structures within the data. Based on the types of multi-view data, existing multi-view clustering methods can be classified into four main categories. (1) Complete multi-view clustering (Yang et al., 2020; Dai et al., 2024; Zhou et al., 2024; Lu et al., 2024b), which is performed when all view information is fully available. (Hu et al., 2020) proposed a dynamic auto-weighted

MVC method with mutual information, optimizing multi-view feature representation by learning view weights. (Yuan et al., 2022) designed a robust self-tuning MVC that addresses initialization sensitivity, cluster number determination, and outlier issues. (2) Incomplete multi-view clustering (Chao et al., 2022; Wen et al., 2024; Wang et al., 2024; Chao et al., 2024), which addresses clustering analysis when some views have missing data. (Zhang et al., 2024b) proposed a framework for incomplete multi-view contrastive clustering that leverages a self-attention mechanism and cross-view prediction. (3) Uncertain multi-view clustering (Liu et al., 2024; Duan, 2020), which is used for clustering data that contains noise, ambiguity, or uncertainty. (Sharma & Seal, 2021) proposed a self-adaptive mixed similarity function, aiming to reduce the effects of outliers and noise. (4) Dynamic multi-view clustering (Yin et al., 2021; Wan et al., 2024; Lu et al., 2024a), which is designed for clustering multi-view data that evolves over time. (Huang et al., 2023) introduced an incremental unsupervised feature selection method to achieve feature selection for incomplete multi-view streaming data.

Although the above methods effectively address issues like missing data, uncertainty, and dynamism in MVC, most assume centralized data storage, overlooking the existence of data silos and the need for privacy protection. To address these issues, FedMVC has been proposed and attracted attention, which can mine complementary clustering structures from multiple clients without exposing private data.

C. Proofs

C.1. Prerequisite Definitions and Lemmas

Definition C.1. *(Kullback-Leibler Divergence).* Let P and Q be probability measures on the same space $\mathcal{X}$, the KL divergence from P to Q is defined as $D_{\mathrm{KL}}(P\|Q) = \int_{\mathcal{X}} P(x) \log \frac{P(x)}{Q(x)} dx$.

Definition C.2. *(Mutual Information).* Let (X, Y) be a pair of random variables defined over the space $\mathcal{X} \times \mathcal{Y}$, with joint distribution $P_{X,Y}$ and marginal distributions P_X and P_Y. The mutual information between X and Y is defined as: $I(X;Y) = D_{\mathrm{KL}}\left(P_{X,Y} \| P_X P_Y\right)$.

Definition C.3. *(Differential Privacy).* let $\mathcal{A}$ be a random mechanism that takes dataset D as input and belongs to set S. Assuming D_1 and D_2 are two neighboring datasets differing in only one point, $\mathcal{A}$ is (ε, δ)-differentially private if:

$$\Pr\left[\mathcal{A}\left(\mathcal{D}_1\right) \in \mathcal{S}\right] \leq \exp(\varepsilon) \cdot \Pr\left[\mathcal{A}\left(\mathcal{D}_2\right) \in \mathcal{S}\right] + \delta, \tag{16}$$

where ε is the privacy budget and δ is the failure probability.

Lemma C.4. *(Hoeffding's Inequality). Let X_i, for $i \in [n]$, be independent random variables with distribution P_X, range $[a, b]$, and $\mathbb{E}[X_i] = \mu$. Let $X = \frac{1}{n}\sum_{i=1}^{n} X_i$ denote the average of the X_i. Then, for any $\lambda \in \mathbb{R}$,*

$$\log \mathbb{E}\left[e^{\lambda(\mu - X)}\right] \leq \frac{\lambda^2 (b-a)^2}{8n} \tag{17}$$

Lemma C.5. *(Donsker-Varadhan Variational Formula). For any measurable, bounded function $h : \Theta \to \mathbb{R}$ we have:*

$$\log \mathbb{E}_{\theta \sim \pi}\left[\mathrm{e}^{h(\theta)}\right] = \sup_{\rho \in \mathcal{P}(\Theta)} \left[\mathbb{E}_{\theta \sim \rho}[h(\theta)] - D_{\mathrm{KL}}(\rho \| \pi)\right] \tag{18}$$

Lemma C.6. *(Gaussian Mechanism (Dwork et al., 2014)). Let $\varepsilon \in (0, 1)$ be arbitrary. The Gaussian mechanism with $\sigma \geq \Delta_2 f \sqrt{2\log(1.25/\delta)}/\varepsilon$ is (ε, δ)-differentially private.*

Lemma C.7. *(Differentially Private SGD (Abadi et al., 2016)). There exist constants c_1 and c_2 so that given the sampling probability $q = \frac{L}{N}$ and the number of steps T, for any $\epsilon < c_1 q^2 T$. The algorithm of DP-SGD is (ε, δ)-differentially private for any $\delta > 0$ if we choose $\sigma \geq c_2 \frac{q\sqrt{T\log(1/\delta)}}{\varepsilon}$.*

C.2. Local Training with Feature Splitting

Recall Eq.(2), we have

$$\begin{aligned}
\mathcal{L}_{\mathbf{X}^m}(w^m) =& -I(\mathbf{Z}_x^m; \mathbf{X}^m) - I(\mathbf{Z}_c^m; \mathbf{Y}) + I(\mathbf{Z}_x^m; \mathbf{Z}_c^m) \\
=& -H(\mathbf{Z}_x^m) - H(\mathbf{X}^m) + H(\mathbf{Z}_x^m; \mathbf{X}^m) - H(\mathbf{Z}_c^m) - H(\mathbf{Y}) + H(\mathbf{Z}_c^m; \mathbf{Y}) + H(\mathbf{Z}_x^m) + H(\mathbf{Z}_c^m) - H(\mathbf{Z}_x^m; \mathbf{Z}_c^m) \\
=& H(\mathbf{Z}_x^m \mid \mathbf{X}^m) + H(\mathbf{Z}_c^m \mid \mathbf{Y}) - H(\mathbf{Z}_x^m; \mathbf{Z}_c^m).
\end{aligned} \tag{19}$$

Then, by constructing the reconstruction network $g_{w_r^m}(\cdot)$ and the clustering layer $g_{w_c^m}(\cdot)$ separately, with w_r^m and w_c^m as learnable parameters, the local loss function is transformed below into a more concrete form:

$$\begin{aligned}\mathcal{L}_{\mathbf{X}^m}(w^m) =& -\int\int p(\mathbf{z}_x^m, \mathbf{x}^m)\log p(\mathbf{z}_x^m \mid \mathbf{x}^m)d\mathbf{z}_x^m d\mathbf{x}^m - \int\int p(\mathbf{z}_c^m, \mathbf{y})\log p(\mathbf{z}_c^m \mid \mathbf{y})d\mathbf{z}_c^m d\mathbf{y} \\ &+ \int\int p(\mathbf{z}_x^m, \mathbf{z}_c^m)\log p(\mathbf{z}_x^m, \mathbf{z}_c^m)d\mathbf{z}_x^m d\mathbf{z}_c^m \\ =& -\mathbb{E}_{p(\mathbf{z}_x^m, \mathbf{x}^m)}\left[\log p\left(\mathbf{z}_x^m \mid \mathbf{x}^m\right)\right] - \mathbb{E}_{p(\mathbf{z}_c^m, \mathbf{y})}\left[\log p\left(\mathbf{z}_c^m \mid \mathbf{y}\right)\right] + \mathbb{E}_{p(\mathbf{z}_x^m, \mathbf{z}_c^m)}\left[\log p\left(\mathbf{z}_x^m, \mathbf{z}_c^m\right)\right] \\ =& \frac{1}{N_m}\sum_{i=1}^{N_m}\|\mathbf{x}_i^m - g_{w_r^m}\left(\mathbf{z}_{i,x}^m\right)\|^2 + \frac{1}{N_m}\sum_{i=1}^{N_m}\sum_{k=1}^{K}\mathbf{y}_{i,k}\log\left(g_{w_c^m}\left(\mathbf{z}_{i,c}^m\right)\right) + \frac{1}{N_m}\sum_{i=1}^{N_m}\log p\left(\mathbf{z}_{i,x}^m, \mathbf{z}_{i,c}^m\right).\end{aligned} \tag{20}$$

C.3. Proof of Lemma 3.1

Lemma C.8. *(Wang et al., 2021) Given an observed dataset $\mathbf{X}^m$, the optimal posterior $p(w^m \mid \mathbf{X}^m)$ for client m update in Eq. (5) should satisfy the following form that*

$$p\left(w^m \mid \mathbf{X}^m\right) = \frac{1}{B}\exp\left\{-\frac{1}{\alpha}U(w^m)\right\}, \tag{21}$$

where B is a normalizing factor that makes the integral of the distribution equal to 1, and $U(w^m)$ is the energy function defined as $U(w^m) = \mathcal{L}_{\mathbf{X}^m}(w^m) - \alpha\log p(w^m)$.

Proof. Recalling the local optimization objective in Eq. (5) for client m, we have

$$\begin{aligned}\min_{p(w^m|\mathbf{X}^m)}\mathcal{L}^m &= \mathbb{E}_{p(w^m|\mathbf{X}^m)}\left[\mathcal{L}_{\mathbf{X}^m}(w^m) + \alpha I\left(w^m; \mathbf{X}^m\right)\right] \\ &= \mathbb{E}_{p(w^m|\mathbf{X}^m)}\left[\mathcal{L}_{\mathbf{X}^m}(w^m)\right] + \alpha\mathbb{E}_{p(\mathbf{X}^m)}\left[D_{\mathrm{KL}}\left[p\left(w^m \mid \mathbf{X}^m\right)\|p(w^m)\right]\right] \\ &= \int p\left(w^m \mid \mathbf{X}^m\right)\mathcal{L}_{\mathbf{X}^m}\left(w^m\right)dw^m + \alpha\int p\left(\mathbf{X}^m\right)\int p\left(w^m \mid \mathbf{X}^m\right)\left[\log p\left(w^m \mid \mathbf{X}^m\right) - \log p\left(w^m\right)\right]dw^m d\mathbf{X}^m.\end{aligned} \tag{22}$$

Differentiating $\mathcal{L}^m$ w.r.t. $p(w^m \mid \mathbf{X}^m)$ results in

$$\nabla_{p(w^m|\mathbf{X}^m)}\mathcal{L} = \mathcal{L}_{\mathbf{X}^m}(w^m) + \alpha\log p\left(w^m \mid \mathbf{X}^m\right) + \alpha - \alpha\log p(w^m). \tag{23}$$

Setting $\nabla_{p(w^m|\mathbf{X}^m)}\mathcal{L}^m = 0$ and solving for $p(w^m \mid \mathbf{X}^m)$ yields

$$\begin{aligned}\log p\left(w^m \mid \mathbf{X}^m\right) &= -\frac{1}{\alpha}\mathcal{L}_{\mathbf{X}^m}(w^m) + \log p(w^m) - 1 \\ p\left(w^m \mid \mathbf{X}^m\right) &= p(w^m)\exp\left\{-\frac{1}{\alpha}\mathcal{L}_{\mathbf{X}^m}(w^m)\right\}\exp\{-1\}.\end{aligned} \tag{24}$$

To integrate the distribution to 1, we add a normalization factor B. Then, we define energy function $U(w^m) = \mathcal{L}_{\mathbf{X}^m}(w^m) - \alpha\log p(w^m)$ and obtain the optimal posterior solution as

$$\begin{aligned}p\left(w^m \mid \mathbf{X}^m\right) &= \frac{1}{B}p(w^m)\exp\left\{-\frac{1}{\alpha}\mathcal{L}_{\mathbf{X}^m}(w^m)\right\} \\ &= \frac{1}{B}\exp\left\{-\frac{1}{\alpha}\left[\mathcal{L}_{\mathbf{X}^m}(w^m) - \alpha\log p(w^m)\right]\right\} \\ &= \frac{1}{B}\exp\left\{-\frac{1}{\alpha}U(w^m)\right\}.\end{aligned} \tag{25}$$

□

C.4. Generalization Analysis

Theorem C.9. *Suppose that $\ell_m\left(w^m, \mathbf{x}_i^m\right)$ for all $m \in M$ is bounded by C and independent, then the expected generalization error satisfies*

$$\mathbb{E}\left[L_{\mathcal{P}}(w) - L_{\mathcal{X}}(w)\right] \leq \frac{C}{M} \sum_{m=1}^{M} \sqrt{\frac{I(w^m; \mathbf{X}^m)}{2N_m}}, \tag{26}$$

where M is the number of participating clients and N_m is the number of samples in the m-th participating client.

Proof. In our settings, we denote the population risk is

$$L_{\mathcal{P}}(w) = \frac{1}{M} \sum_{m=1}^{M} L_{p(\mathbf{X}^m)}(w^m) = \frac{1}{M} \sum_{m=1}^{M} \mathbb{E}_{\mathbf{X}^m \sim p(\mathbf{X}^m)}\left[\ell_m\left(w^m, \mathbf{X}^m\right)\right], \tag{27}$$

and define the empirical risk is

$$L_{\mathcal{X}}(w) = \frac{1}{M} \sum_{m=1}^{M} L_{\mathbf{X}^m}(w^m) = \frac{1}{M} \sum_{m=1}^{M} \left[\frac{1}{N_m} \sum_{i=1}^{N_m} \ell_m\left(w^m, \mathbf{x}_i^m\right)\right]. \tag{28}$$

Then, the generalization error can be written as

$$\mathbb{E}\left[L_{\mathcal{P}}(w) - L_{\mathcal{X}}(w)\right] = \frac{1}{M} \sum_{m=1}^{M} \mathbb{E}_{p(\mathbf{X}^m)} \mathbb{E}_{p(w^m|\mathbf{X}^m)}\left[L_{p(\mathbf{X}^m)}(w^m) - L_{\mathbf{X}^m}(w^m)\right], \tag{29}$$

where $\mathbb{E}_{p(\mathbf{X}^m)} \mathbb{E}_{p(w^m|\mathbf{X}^m)}\left[L_{p(\mathbf{X}^m)}(w^m) - L_{\mathbf{X}^m}(w^m)\right]$ can be regarded as the expected generalization error of any participating client m.

Lemma C.10. *(Generalization Bound for client m). Suppose that $\ell_m\left(w^m, \mathbf{x}_i^m\right)$ is bounded by C, then the expected generalization error for client m satisfies*

$$\mathbb{E}_{p(\mathbf{X}^m)} \mathbb{E}_{p(w^m|\mathbf{X}^m)}\left[L_{p(\mathbf{X}^m)}(w^m) - L_{\mathbf{X}^m}(w^m)\right] \leq \sqrt{\frac{C^2 I(w^m; \mathbf{X}^m)}{2N_m}}, \tag{30}$$

where $p(\mathbf{X}^m)$ denotes a distribution over examples in client m and N_m is the number of samples for client m.

Proof. For client m, its generalization error can be expressed as

$$L_{p(\mathbf{X}^m)}(w^m) - L_{\mathbf{X}^m}(w^m) = \mathbb{E}_{\mathbf{X}^m \sim p(\mathbf{X}^m)}\left[\ell_m\left(w^m, \mathbf{X}^m\right)\right] - \frac{1}{N_m} \sum_{i=1}^{N_m} \ell_m\left(w^m, \mathbf{x}_i^m\right) \tag{31}$$

By recalling Hoeffding's inequality in Lemma C.4 and suppose that $\ell_m\left(w^m, \mathbf{x}_i^m\right)$ is bounded by C, we have

$$\mathbb{E}_{p(\mathbf{X}^m)}\left[e^{\lambda\left[L_{p(\mathbf{X}^m)}(w^m) - L_{\mathbf{X}^m}(w^m)\right]}\right] \leq \exp\left(\frac{\lambda^2 C^2}{8N_m}\right). \tag{32}$$

Then, we apply the Donsker-Varadhan variational formula in Lemma C.5, set $\mathcal{P}(\Theta)$ denotes the set of all probability distributions, and $\pi \triangleq p(w)$ is a prior distribution over hypothesis. Then we obtain

$$\begin{aligned} &\mathbb{E}_{p(\mathbf{X}^m)}\left(\sup_{\rho \in \mathcal{P}(\Theta)} \mathrm{e}^{\lambda\left[L_{p(\mathbf{X}^m)}(w^m) - L_{\mathbf{X}^m}(w^m)\right] - D_{\mathrm{KL}}(\rho\|\pi)}\right) \leq \mathrm{e}^{\frac{\lambda^2 C^2}{8N_m}}, \\ &\mathbb{E}_{p(\mathbf{X}^m)}\left(\sup_{\rho \in \mathcal{P}(\Theta)} \mathrm{e}^{\lambda\left[L_{p(\mathbf{X}^m)}(w^m) - L_{\mathbf{X}^m}(w^m)\right] - D_{\mathrm{KL}}(\rho\|\pi) - \frac{\lambda^2 C^2}{8N_m}}\right) \leq 1. \end{aligned} \tag{33}$$

Further, by replacing $p(w^m \mid \mathbf{X}^m)$ with ρ, we have

$$\mathbb{E}_{p(\mathbf{X}^m)}\mathbb{E}_{w^m\sim p(w^m|\mathbf{X}^m)}\left[L_{p(\mathbf{X}^m)}(w^m) - (L_{\mathbf{X}^m}(w^m) + \frac{D_{\mathrm{KL}}(p(w^m \mid \mathbf{X}^m)\|\pi)}{\lambda} + \frac{\lambda C^2}{8N_m})\right] \leq 0, \tag{34}$$

$$\begin{aligned}\mathbb{E}_{p(\mathbf{X}^m)}\mathbb{E}_{w^m\sim p(w^m|\mathbf{X}^m)}\left[L_{p(\mathbf{X}^m)}(w^m) - (L_{\mathbf{X}^m}(w^m)\right] &\leq \mathbb{E}_{p(\mathbf{X}^m)}\mathbb{E}_{w^m\sim p(w^m|\mathbf{X}^m)}\left[\frac{D_{\mathrm{KL}}(p(w^m \mid \mathbf{X}^m)\|\pi)}{\lambda} + \frac{\lambda C^2}{8N_m}\right] \\ &\leq \frac{I(w^m;\mathbf{X}^m)}{\lambda} + \frac{\lambda C^2}{8N_m}.\end{aligned} \tag{35}$$

When $\lambda = \sqrt{\frac{8N_m I(w^m;\mathbf{X}^m)}{C^2}}$, we get

$$\mathbb{E}_{p_{(\mathbf{X}^m)}}\mathbb{E}_{p(w^m|\mathbf{X}^m)}\left[L_{p(\mathbf{X}^m)}(w^m) - L_{\mathbf{X}^m}(w^m)\right] \leq \sqrt{\frac{C^2 I(w^m;\mathbf{X}^m)}{2N_m}}. \tag{36}$$

□

Based on Eq. (29) and Eq. (30), we have

$$\mathbb{E}\left[L_{\mathcal{P}}(w) - L_{\mathcal{X}}(w)\right] \leq \frac{C}{M}\sum_{m=1}^{M}\sqrt{\frac{I(w^m;\mathbf{X}^m)}{2N_m}}. \tag{37}$$

□

C.5. Privacy Analysis

Our privacy analysis contains two steps, saying, differential privacy for each client and the overall analysis for ESFMC.

C.5.1. Differential Privacy for Each Client

Our proposed approach provides client-level privacy protection through posterior inference based on SGLD sampling and feature splitting, using client m as an example for analysis. Firstly, under the Gaussian mechanism in Lemma C.6, SGLD in the local posterior inference has been demonstrated to provide strict differential privacy.

Lemma C.11. *(Differentially Private SGLD). Assuming that $\ell_m\left(w^m, \mathbf{x}_i^m\right)$ is L-smooth, when $k \geq \mathcal{O}\left(\frac{\sqrt{\alpha}\epsilon^2}{\log(2/\delta)}\right)$, the algorithm in client m preserves (ε, δ)-differential privacy.*

Proof. Based on the Gaussian mechanism in Lemma C.6, we attain $\mathcal{O}\left(\sqrt{2\log(1.25/\delta)}/\varepsilon\right)$ for selecting σ. After rearranging, we have $\varepsilon = \sqrt{2\log(1.25/\delta)}/\sigma$ for further analysis. Previous work has shown that under this Gaussian mechanism, in standard SGLD, $k \geq \frac{\varepsilon^2 N_m}{32\tau\log(2/\delta)}$ ensures the privacy loss to be smaller than $\frac{\varepsilon\sqrt{N_m}}{\sqrt{32\tau k\log(2/\delta)}}$ with probability $> 1 - \frac{\tau\delta}{2N_m k}$, where k is the local epoch, N_m is the number of samples in client m and τ represents the samples sampled in current epoch. Additionally, we have an extra parameter $\sqrt{\alpha}$ in the noise variance. That is, we need $k \geq \frac{\sqrt{\alpha}\varepsilon^2 N_m}{32\tau\log(2/\delta)}$ for preserving (ε, δ)-differential privacy in client m. □

We also use differential privacy to analyze the feature-splitting strategy for preserving privacy, demonstrating that sharing the clustering-related features $\mathbf{Z}_c^m$ provides stronger privacy protection than sharing the latent features $\mathbf{Z}^m$ directly.

Lemma C.12. *For sharing clustering-related features $\mathbf{Z}_c^m$, (ε, δ)-differential privacy in client m if $\varepsilon = \mathcal{O}\left(\rho_c\sqrt{k\log(1/\delta)}/\sigma_c\right)$. Nevertheless, For sharing latent features $\mathbf{Z}^m$, (ε', δ)-differential privacy in client m if $\varepsilon' = \mathcal{O}\left(\sqrt{k\log(1/\delta)}\left(\rho_c/\sigma_c + \rho_x/\sigma_x\right)\right)$.*

Proof. To facilitate the privacy analysis, we add the noise by the similar way to DP-SGD (Abadi et al., 2016), i.e., employ the idea of the ℓ_2-norm clipping relating to the selection of the noise level σ. Specifically, We apply the random mechanism with noise distribution $\mathcal{N}\left(\mathbf{0}, \sigma_c^2\mathbf{I}\right)$ to $\mathbf{Z}_c^m$ and $\mathcal{N}\left(\mathbf{0}, \sigma_x^2\mathbf{I}\right)$ to $\mathbf{Z}_x^m$. Recall Eq.(3), we obtain $\mathbf{Z}_c^m$ and $\mathbf{Z}_x^m$ from latent features $\mathbf{Z}^m$ by designing two adaptive feature projection layers, so we define $\mathbf{Z}_c^m = \rho_c\mathbf{Z}^m$ and $\mathbf{Z}_x^m = \rho_x\mathbf{Z}^m$. Building on Lemma C.7, we attain $\mathcal{O}(\rho_c\sqrt{k\log(1/\delta)}/\varepsilon)$ and $\mathcal{O}(\rho_x\sqrt{k\log(1/\delta)}/\varepsilon)$ for selecting σ_c and σ_x, respectively, where k corresponds to T in the original equation, denoting times of using private data for differentially private training. By rearranging variables, we have $\varepsilon_c = \rho_c\sqrt{R\log(1/\delta)}/\sigma_c$ and $\varepsilon_x = \rho_x\sqrt{R\log(1/\delta)}/\sigma_x$. Since $\mathbf{Z}_c^m + \mathbf{Z}_x^m = \rho_c\mathbf{Z}^m + \rho_x\mathbf{Z}^m$, $\mathbf{Z}^m$ can be viewed as a combination of $\mathbf{Z}_c^m$ and $\mathbf{Z}_x^m$, by combining ε_c and ε_c, we attain $\varepsilon' = \mathcal{O}\left(\sqrt{k\log(1/\delta)}\left(\rho_c/\sigma_c + \rho_x/\sigma_x\right)\right)$. Since $\mathbf{Z}_x^m$ are kept local, adversaries cannot access them, effectively adding large noise ($\sigma_x \to \infty$). Our analysis shows that sharing $\mathbf{Z}_c^m$ instead of $\mathbf{Z}^m$ offers stronger privacy guarantees, leading to a smaller privacy budget ε. □

Combining Lemma C.11 and Lemma C.12, we have the following theorem.

Theorem C.13. *Assuming that $\ell_m\left(w^m, \mathbf{x}_i^m\right)$ is L-smooth, when $k \geq \mathcal{O}\left(\frac{\sqrt{\alpha}\epsilon^2}{\log(2/\delta)}\right)$, our proposed ESFMC preserves client-level (ε, δ)-differential privacy, where $\varepsilon = (\sqrt{2\log(1.25/\delta)}/\sigma + \rho_c\sqrt{k\log(1/\delta)}/\sigma_c)$.*

C.5.2. Privacy Analysis for ESFMC

In addition to privacy analysis for each client, we conduct the overall privacy analysis of the proposed ESFMC by composition theorem.

Lemma C.14. *(Composition Theorem (Kairouz et al., 2015)). For any $\varepsilon > 0$, $\delta \in [0, 1]$, and $\tilde{\delta} \in [0, 1]$, the class of (ε, δ)-differentially private mechanisms satisfies $\left(\tilde{\varepsilon}_{\tilde{\delta}}, 1 - (1-\delta)^M(1-\tilde{\delta})\right)$-differential privacy under M-fold adaptive composition for $\tilde{\varepsilon}_{\tilde{\delta}} = \min\left\{M\varepsilon, \frac{(e^\varepsilon-1)\varepsilon M}{e^\varepsilon+1} + \varepsilon\sqrt{2M\log\left(e + \frac{\sqrt{M\varepsilon^2}}{\tilde{\delta}}\right)}, \frac{(e^\varepsilon-1)\varepsilon M}{e^\varepsilon+1} + \varepsilon\sqrt{2M\log\left(\frac{1}{\tilde{\delta}}\right)}\right\}$.*

The composition theorem indicates that when the interactive mechanisms are individually differentially private, their concurrent composition maintains privacy parameters (in terms of pure or approximate differential privacy). By combining Lemma C.13 and Lemma C.14, we derive Theorem 4.2.

D. Experimental Details

D.1. Datasets

We conduct the experiments on the following public datasets.

- **Caltech** (Fei-Fei et al., 2004) is an RGB image dataset that provides multiple views including 40-dim wavelet moments (WM), 254-dim CENsus TRansform hISTogram (CENTRIST), 928-dim Local Binary Patterns (LBP), 512-dim GeneralIzed Search Trees (GIST), and 1984-dim Histogram of Oriented Gradients (HOG) features. We adopt 200 samples from each class and 1400 samples in total for evaluating the robustness of the multi-view clustering methods.
- **HW** are represented by six kinds of features extracted from its binary image. Each class has 200 samples. Each instance has six visual views, including profile correlations, Fourier coefficients of the character shapes, Karhunen-Love, morphological features, pixel averages in 2 × 3 windows, and Zernike moments.
- **MNIST-USPS** (Peng et al., 2019) is a widely-used dataset for handwritten digits (0-9) and consists of 5000 examples with two views of digital images. The MNIST feature size is 28 × 28, while the USPS is 16 × 16.
- **Synthetic3d** (Kumar et al., 2011) comprises 3 categories, containing a total of 600 samples, each with three views, and each view having a dimensionality of 3.
- **BDGP** (Cai et al., 2012) comprises 2500 examples related to drosophila embryos, each represented by a 1750-dimensional visual feature and a 79-dimensional textual feature. GIST and LBP features
- **Scene** (Fei-Fei & Perona, 2005) includes 4,485 images in 15 classes, each sample has three views: 20-dim GeneralIzed Search Trees (GIST), 59-dim Histogram of Oriented Gradients (HOG) and 40-dim Local Binary Patterns (LBP).

D.2. Implementation Details

The models of all methods are implemented on the PyTorch (Paszke et al., 2019) platform using NVIDIA RTX-3090 GPUs. The activation function is ReLU (Glorot et al., 2011). For all the datasets used, the learning rate is fixed at 0.001. Before conducting the first communication round, clients have not received global clustering results to assist in feature splitting, we use local clustering results to assist in splitting in this round. In subsequent communication rounds, local training is conducted for 300 epochs for all datasets on each client. The communication rounds between the server and clients are set to $R = 15$ for Caltech, MNIST-USPS, and Synthetic3d datasets, and $R = 5$ for other datasets.

D.3. Comparison Methods

To demonstrate the performance of our proposed ESFMC, we select several relevant algorithms as comparison methods. These include five centralized multi-view clustering methods:

- GIMC-FLSD (2020) (Wen et al., 2020) handles incomplete data through graph-regularized matrix factorization and semantic consistency constraints.
- HCP-IMSC (2022) (Li et al., 2022) uses tensor decomposition regularization to preserve high-order view and sample correlations.
- IMVC-CBG (2022) (Wang et al., 2022) utilizes anchor learning and a bipartite graph framework to achieve efficient incomplete multi-view clustering (IMVC).
- DSIMVC (2022) (Tang & Liu, 2022) uses a bi-level optimization framework to reduce the impact of semantic inconsistency on IMVC performance.
- AGDIMC (2024) (Pu et al., 2024) is an IMVC method that combines view-specific deep encoders with partial latent graphs for adaptive feature imputation.

Four federated multi-view clustering methods are included:

- FedDMVC (2023) (Chen et al., 2023) is a FedMVC that concurrently tackles feature heterogeneity and IMVC.
- FedMVFCM (2023) (Hu et al., 2023) primarily focuses on achieving FedMVC through the sharing and aggregation of cluster centroids.
- FedMVFPC (2023) (Hu et al., 2023) uses federated learning to address the challenges of multi-view fuzzy clustering on distributed devices.
- FCUIF (2024) (Ren et al., 2024) introduces adaptive alignment and unsupervised imputation techniques to address the challenges of unaligned and incomplete multi-view data.

E. Additional Experiment Results

E.1. Communication Overhead

Table 4. Runtime and communication overhead by ESFMC.

Dataset	Runtime	Communication overhead
Caltech	703.6s	2.5KB
HW	617.2s	1.4KB
MNIST-USPS	1199.5s	3.6KB
Synthetic3d	296.2s	0.7KB
BDGP	265.4s	0.6KB
Scene	450.5s	1.6KB

We report some results on the runtime and communication overhead by ESFMC to give the reader some information about the computational resources used by the method. Table 4 shows that runtime and communication overhead by ESFMC is small and easy to reproduce.

E.2. Ablation Study about Shared Features

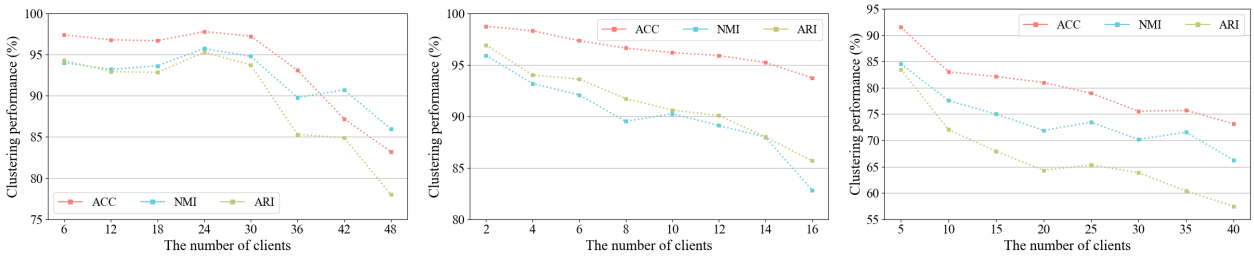
We also explore the impact of sharing different feature types on clustering performance on Caltech dataset. The results indicate in Table 5, while sample-related and complete features reveal partial clustering structures, the clustering-related features obtained through feature splitting exhibit superior accuracy and effectiveness in mining clustering structures, thereby validating the efficacy of our feature-splitting strategy.

Table 5. Ablation study about shared features on Caltech dataset.

Types of sharing features	ACC	NMI	ARI
Sample-related features	79.93	73.27	66.82
Clustering-related features (ESFMC)	91.50	84.54	83.45
Complete features	87.29	77.86	74.79

E.3. Extending to Cross-Device Scenarios

Our proposed method operates under the assumption that M views are distributed across M clients, with each client holding data for a unique view. Several existing FedMVC methods (Huang et al., 2022; Chen et al., 2023) also follow this assumption. So our approach is particularly suitable for cross-silo scenarios with a limited number of clients, such as hospitals or large institutions. Luckily, it can also be extended to cross-device scenarios involving numerous clients and has shown promising performance on certain datasets.



(a) HW (b) BDGP (c) Caltech

Figure 4. Clustering performance with the varying number of clients.

Figure 4 illustrates the impact of the number of clients on clustering performance without sharing any additional information. Notably, increasing the number of clients reduces the number of samples per client, which can hinder the feature-splitting process during local training, leading to suboptimal clustering-related feature extraction. The results show that ESFMC maintains stable performance on the HW and BDGP datasets when the number of clients increases moderately. However, with a significant increase, the sample size per client becomes insufficient for effective training, resulting in a sharp decline in clustering performance. To adapt ESFMC to the Caltech dataset and scenarios with a large number of clients, sharing partial local model information may be necessary. While this strategy could help maintain clustering performance, it introduces new privacy concerns that require further investigation.