Information Fusion 108 (2024) 102357

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

A novel federated multi-view clustering method for unaligned and incomplete data fusion

Yazhou Ren [a,b,*], Xinyue Chen [a,*], Jie Xu [a], Jingyu Pu [a], Yonghao Huang [a], Xiaorong Pu [a,b], Ce Zhu [c], Xiaofeng Zhu [a,b], Zhifeng Hao [d], Lifang He [e]

[a] School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, 611731, China
[b] Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, Shenzhen, 518000, China
[c] School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu, 611731, China
[d] College of Science, Shantou University, Shantou, 515063, China
[e] Department of Computer Science and Engineering, Lehigh University, Bethlehem, 18015, PA, USA

ARTICLE INFO

Keywords:
Multi-view clustering
Federated learning
Data fusion

ABSTRACT

Recently, federated multi-view clustering (FedMVC) has emerged as a powerful tool to uncover complementary cluster structures across distributed clients, gaining significant attention in the realm of data fusion. While FedMVC methods have adeptly addressed the challenges of feature heterogeneity among various clients, achieving notable success in controlled environments. Their applicability often hinges on the assumptions of strict alignment and data completeness across multi-view clients. These assumptions, unfortunately, are not always consistent with real-world conditions. Specifically, practical applications often come with (1) unaligned multi-view data and (2) missing data. Current FedMVC methods struggle to effectively address these challenges. To bridge this gap, this paper presents FCUIF, a novel method that eliminates the need for data alignment and completeness assumptions. To tackle unaligned data, FCUIF leverages both sample commonality and view versatility to adaptively generate alignment matrices, ensuring effective cross-view alignment. For the challenge of missing data, FCUIF uses an unsupervised technique to evaluate and refine imputation quality, efficiently handling various scenarios of incomplete multi-view data. Our extensive experiments using four public datasets demonstrate FCUIF's superior performance when dealing with unaligned and incomplete multi-view data. The code is available at https://github.com/5Martina5/FCUIF.

1. Introduction

Federated learning (FL), is a widely adopted paradigm in distributed machine learning, enabling multiple clients to collaboratively train models without compromising privacy [1–4]. Most existing FL methods usually assume that each client's private data belong to the same modality or view, while exploring the problems of non-iid data, privacy, and communication cost issues [5–7]. However, with the advancement of sensing technology and the dramatic increase of multi-view/modal data in recent years, the study of multi-source and heterogeneous data fusion has become an emerging trend in FL [8–11]. Federated multi-view learning addresses the issue of feature heterogeneity across different views while guaranteeing privacy. It facilitates knowledge sharing among multiple clients and is useful in various fields, including recommendation [12], medical prediction [13], and clustering analysis [14].

Clustering analysis with federated multi-view learning, also referred to as federated multi-view clustering (FedMVC), addresses feature heterogeneity and uncovers complementary cluster structures across multiple clients. On the one hand, it tackles the issue of feature heterogeneity in multi-view data across multiple clients by proposing distinct data fusion strategies on the server; on the other hand, it leverages collaborative training among multiple clients and their data interactions with the server to mine the private and common information across multiple views, thereby identifying complementary clustering structures.

While some FedMVC methods have shown promising results, they overlook two critical issues. Firstly, the success of existing FedMVC methods [14,15] heavily relies on the assumption that multi-view data must be strictly aligned across clients. However, in real-world scenarios, it is unlikely that different clients collect samples in the same order. Note that when using FedMVC methods under such situations, it would be necessary to expose sample IDs for cross-client alignment, which compromises privacy in the FL scenario. Secondly, there is limited research addressing incomplete multi-view data in distributed

* Corresponding authors.
E-mail addresses: yazhou.ren@uestc.edu.cn (Y. Ren), martinachen2580@gmail.com (X. Chen).

https://doi.org/10.1016/j.inffus.2024.102357
Received 30 September 2023; Received in revised form 26 February 2024; Accepted 12 March 2024
Available online 16 March 2024
1566-2535/

environments. Although there has been significant progress in handling incomplete multi-view clustering (IMVC) in centralized settings using imputation or fusion techniques [16–18], there is a lack of discussions about strategies for such data. Additionally, the challenge of imbalanced incomplete data across clients in the FL context needs further exploration.

To address the aforementioned two issues, we introduce a novel FedMVC method, named FCUIF (Federated multi-view Clustering method for Unaligned and Incomplete data Fusion). FCUIF is designed to address the challenges of unaligned and incomplete data in multi-view datasets across distributed clients. For example, medical tests distributed across different healthcare institutions can be considered as different views. Most patients do not select healthcare institutions in the same order, leading to unaligned data. Additionally, patients do not select all institutions to undergo tests, resulting in incomplete data. FCUIF can enable collaboration and clustering for unaligned and incomplete data while preserving privacy. Specifically, to tackle unaligned data, FCUIF designs an adaptive alignment module, which achieves cross-client view adaptation without exposing additional information like sample IDs. This module utilizes extracted sample commonality features to capture data similarity across clients, thus adaptively computing alignment matrices and assisting in obtaining aligned global features. For the issue of missing data, FCUIF proposes an adaptive imputation module to effectively assess imputation quality in an unsupervised manner, achieving adaptive imputation on the server to address the issue of incomplete multi-view data across clients. This module leverages sample commonality and view versatility to adaptively impute unavailable parts across clients, effectively handling various incomplete data scenarios in distributed environments, including point, block, and imbalanced incomplete data situations.

In summary, the proposed method facilitates the fusion of different view data and mines complementary clustering structures within the FL scenario. The general framework of FCUIF is shown in Fig. 1. In the client environment, multiple clients extract view-specific embedded features and cluster assignments from local private data using deep autoencoders and global self-supervised information, which are then uploaded to the server. In the server environment, the server extracts sample commonality and view versatility from the data uploaded by each client, and utilizes the adaptive alignment module and adaptive imputation module to reconstruct the cross-view relationship between samples. Additionally, the server yields global self-supervised information that assists local training at each client and facilitates the mining of high-quality global clustering structures.

This paper presents substantial enhancements and expansions over our prior work, FedDMVC [14], addressing several crucial dimensions. Firstly, in FedDMVC, our focus was limited to surmounting two principal challenges of FedMVC: feature heterogeneity and incomplete multi-view data in distributed contexts. However, our current work, FCUIF, delves deeper by additionally addressing alignment challenges present in multi-view datasets dispersed across distributed clients. Secondly, FedDMVC employed global prototypes and view-specific patterns to address the issue of incomplete multi-view data in distributed environments. Notably, this approach largely depended on imputation, which had its limitations. Mainly, it could not evaluate the quality of imputed data, leaving it vulnerable to the detriments of low-quality imputed features. FCUIF, on the other hand, introduces an unsupervised methodology, effectively gauging imputation quality and employing adaptive imputation strategies for data expansion. Lastly, FedDMVC was designed to handle point-wise incomplete data in multi-view datasets within distributed settings. In contrast, FCUIF exhibits greater flexibility by managing various forms of incomplete data scenarios, encompassing point-wise, block-wise, and imbalanced incomplete data. Overall, the main contributions of this work are summarized as follows:

- We propose a novel FedMVC method that can address the issues of unaligned and incomplete data in multi-view datasets across distributed clients. Additionally, it designs a data fusion strategy on the server to mine high-quality global clustering structures.
- Our method is based on sample commonality and view versatility, allowing the server to adaptively calculate alignment matrices for cross-view alignment, and evaluate imputation quality, resulting in an adaptive imputation technique that exhibits robustness.
- Our method effectively addresses various unaligned and incomplete data scenarios in distributed environments. Extensive experiments on public datasets demonstrate its superior performance in terms of generality and clustering effectiveness.

2. Related work

2.1. Multi-view clustering

Multi-view clustering (MVC) methods leverage consistency and complementary information between multiple views to enhance clustering effectiveness. For most MVC methods, their feasibility is based on the foundation of data integrity, which means multi-view data must be complete and aligned. From this perspective, existing MVC methods can be categorized into three classes based on the distributions or characteristics of multi-view data. (1) Traditional multi-view clustering [19–26], which uncovers hidden patterns and structures by leveraging complete multi-view data for clustering. For example, Zhou et al. [25] proposed a novel multi-view subspace clustering approach that exploits the underlying correlations from multiple views while capturing view-specific information from each independent view. Ming et al. [26] assumed that multi-view data shares a common latent embedding and proposed a novel MVC method by learning a shared generative latent representation. Both methods utilize the complete multi-view information to learn an informative and consistent representation of data. (2) Incomplete multi-view clustering [17,27–30], which leverages the complete views to predict the missing data. Lin et al. [17] reconstructed missing views by minimizing the conditional entropy of multiple views using dual prediction. Liu et al. [28] imputed each incomplete base matrix generated by incomplete views with a learned consensus clustering matrix. (3) Unaligned multi-view clustering [31,32], in which case the samples of the same instance are unaligned. Huang et al. [31] attempted to align the data by establishing the cross-view correspondence at the instance level in an unsupervised manner. Although Yang et al. [32] addressed unaligned and incomplete multi-view clustering tasks in a centralized setting, there is no further exploration of data privacy issues in federated environments, as well as cross-view relationship mining.

However, the aforementioned MVC methods can only separately address data unaligned and incomplete data issues in centralized environments, these methods also do not translate well to distributed environments. Although some distributed MVC methods [33,34] have been proposed, they cannot effectively address the unique issues introduced by federated learning, such as feature heterogeneity and cross-client privacy dilemma. Furthermore, the solutions to incomplete multi-view data in distributed environments have not been well-researched. In this paper, we propose a novel FedMVC method that can address the issues of unaligned and incomplete data in multi-view datasets across distributed clients, additionally evaluate imputation quality to enhance adaptability and robustness.

2.2. Federated multi-view learning

Federated multi-view learning (FedMVL) is a decentralized approach that conducts multi-view learning within a federated setting. Existing FedMVL methods can be categorized based on the downstream tasks they address. One of the primary tasks involves clustering analysis across multiple clients. For instance, Huang et al. [15] were

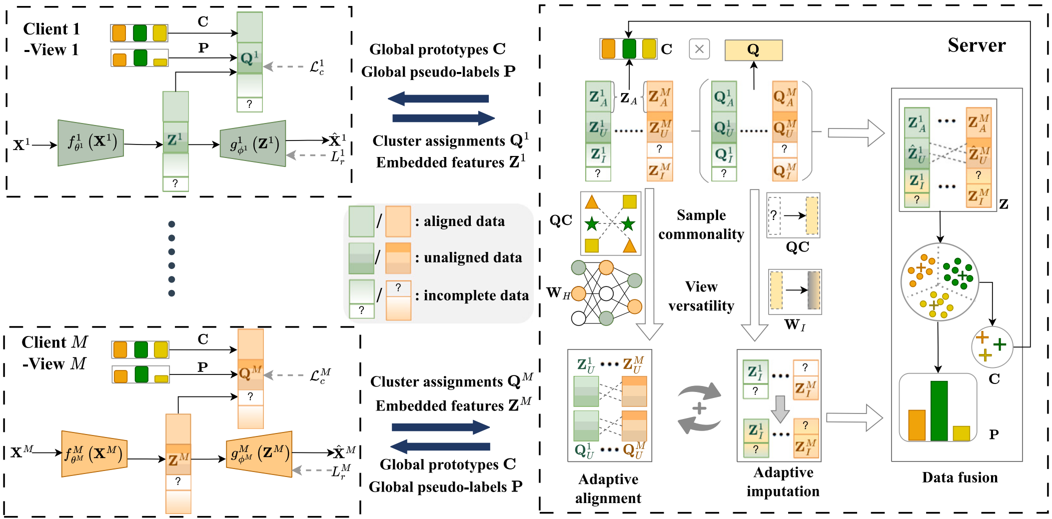


Fig. 1. An overview of the proposed FCUIF framework. It contains M clients and a server. (1) Clients: Multiple clients integrate global self-supervised information to extract view-specific embedded features and cluster assignments from their local private data. (2) Server: The server employs adaptive alignment and imputation modules to reconstruct cross-view relationships among samples and utilize a data fusion strategy to mine the global clustering structures.

the first to consider challenges such as high communication costs, fault tolerance, and issues related to stragglers in distributed multi-view learning. We introduced an innovative federated deep multi-view clustering method in our previous work, FedDMVC [14]. This method effectively mines complementary cluster structures from distributed multi-view data while addressing challenges such as data privacy and feature heterogeneity. Another primary application of FedMVL is in recommendation systems. For example, Flanagan et al. [35] proposed a matrix-factorization-based federated learning framework for personalized recommendations. Huang et al. [12] proposed a versatile content-based federated multi-view framework tailored for recommendation scenarios, effectively mitigating the cold-start problem. Moreover, FedMVL can also be applied to classification problems in specific domains. Xu et al. [36] proposed a late fusion approach to tackle the issue of inconsistent time series in multi-view data for diagnosing depression. Che et al. [13] introduced a general multi-view learning framework that leverages the federated learning paradigm to securely share medical data among institutions while preserving privacy, leading to improved classification results. Recently, research on distributed algorithms under large-scale pre-trained models has gained increasing attention. These methods achieve competitive performance with minimal learnable parameters, providing further insights into the development of the FedMVL field. For example, Feng et al. [37] proposed learning joint visual cues in the null space of globally informed hints reconstructed from magnetic resonance imaging data. Feng et al. [38] enabled instance-adaptive inference by scale and shift deep features upon a pre-trained model, to handle intra-client data heterogeneity.

Currently, many FedMVL methods have achieved excellent results by designing suitable frameworks tailored to the different distributions or characteristics of multi-view data. However, these methods exhibit significant limitations in unsupervised multi-view environments, as they primarily focus on labeled data. Additionally, nearly all FedMVL methods operate within the context of complete multi-view information, which is not easily obtainable due to data storage constraints and the involvement of multiple clients. Furthermore, many methods have failed to consider the unaligned multi-view data, which is also quite common in real-world scenarios. In contrast to previous methods, we propose an adaptive alignment model to address the issue of unaligned data. Moreover, we design an adaptive unsupervised imputation module to handle the problem of incomplete multi-view data across clients.

3. Methodology

In this study, our objective is to address the challenges posed by unaligned and incomplete data in multi-view datasets dispersed across distributed clients, ultimately aiming to mine high-quality global clustering structures.

3.1. Problem formulation

Building upon previous work [14,15], we consider a multi-view dataset with M views, denoted as $\left\{\mathbf{X}^m \in \mathbb{R}^{N_m \times D_m}\right\}_{m=1}^{M}$, distributed across M clients. For client m, its local private data with mth view have dimension D_m and sample size N_m. Note that due to the uncertainty in data collection, each client exhibits variations in terms of sample order, sample count, and sample features. Given these characteristics, we further partition the local private data into $\mathbf{X}^m = \left[\mathbf{X}_A^m; \mathbf{X}_U^m; \mathbf{X}_I^m\right]$, where $\mathbf{X}_A^m$ represents the part of client m's data that can be aligned across views with other clients by information such as sample IDs. Meanwhile, $\mathbf{X}_U^m$ comprises the parts of client m's data that overlap with other clients' data but cannot be aligned across views without compromising privacy. Lastly, $\mathbf{X}_I^m$ denotes the unique samples within client m's data, which do not pose unaligned challenges as they are exclusive to client m.

3.2. Model architecture

Our model architecture consists of M clients and a server. Each client conducts local view-specific training using its private data, i.e., client m trains using its local data $\mathbf{X}^m \in \mathbb{R}^{N_m \times D_m}$. Meanwhile, the server utilizes the embedded features $\{\mathbf{Z}^m\}_{m=1}^{M}$ and cluster assignments $\{\mathbf{Q}^m\}_{m=1}^{M}$ obtained from all clients to extract cross-client information and perform global data fusion training.

3.2.1. Local view-specific training

We adopt the same approach to construct a local autoencoder and clustering layer for each client. This allows us to extract view-specific embedded features and cluster assignments unique to each client, which are then uploaded to the server. Additionally, we incorporate a global perspective into local model training by referencing global prototypes $\mathbf{C}$, and global pseudo-labels $\mathbf{P}$, obtained from the server. Importantly, these embedded features and cluster assignments do not expose the raw data of individual clients, and privacy is ensured through the utilization of techniques such as homomorphic encryption [39] or secure multi-party computation [40] to mitigate model inversion attacks. We analyze the local training process of client m as follows.

Deep autoencoders have found widespread use in extracting high-level representations from raw features [41,42]. Therefore, we design a view-specific encoder $f^m_{\theta^m}$, and its corresponding decoder $g^m_{\phi^m}$ for client m, where θ^m and ϕ^m represent the learnable parameters. The local data $\mathbf{X}^m$ are encoded into $\mathbf{Z}^m \in \mathbb{R}^{N_m \times d_m}$ using $f^m_{\theta^m}$, and then $\mathbf{Z}^m$ are reconstructed as $\hat{\mathbf{X}}^m$ using $g^m_{\phi^m}$, where $\hat{\mathbf{X}}^m$ represents the reconstructed samples generated by the autoencoder. Client m employs a deep autoencoder to project its data into a low-dimensional d_m space. This process preserves the privacy of the original data while capturing informative latent features, and it can be achieved by minimizing the following reconstruction loss:

$$\mathcal{L}^m_r = \left\|\mathbf{X}^m - g^m_{\phi^m}\left(\mathbf{Z}^m\right)\right\|^2_F = \sum_{i=1}^{N_m} \left\|\mathbf{x}^m_i - g^m_{\phi^m}\left(f^m_{\theta^m}\left(\mathbf{x}^m_i\right)\right)\right\|^2_2. \tag{1}$$

To obtain cluster assignments, we construct a clustering layer $c^m_{\mathbf{u}^m}$ with learnable parameters $\left\{\mathbf{u}^m_j \in \mathbb{R}^{d_m}\right\}_{j=1}^K$, based on existing deep embedding clustering methods [43,44]. Here, K represents the number of clustering targets. After initializing this clustering layer with global prototypes $\mathbf{C}$, we obtain soft cluster assignments $\mathbf{Q}^m \in \mathbb{R}^{N_m \times K}$. Specifically, the assignment of the ith embedded feature $\mathbf{z}^i_m$ to the jth cluster is expressed as:

$$q^m_{ij} = c^m_{\mathbf{u}^m}\left(\mathbf{z}^m_i\right) = \frac{\left(1 + \left\|\mathbf{z}^m_i - \mathbf{u}^m_j\right\|^2_2\right)^{-1}}{\sum_{j=1}^K \left(1 + \left\|\mathbf{z}^m_i - \mathbf{u}^m_j\right\|^2_2\right)^{-1}}. \tag{2}$$

For client m, we transform the global pseudo-labels $\mathbf{P}$ into localized supervised information $\mathbf{P}^m$ through a mapping function $\mathcal{F}_m(\mathbf{P}) : \mathbf{P} \in \mathbb{R}^{N \times K} \longmapsto \mathbf{P}^m \in \mathbb{R}^{N_m \times K}$, where N represents the total number of samples across all clients. Additionally, we optimize the clustering loss between the pseudo-labels $\mathbf{P}^m$ and its own cluster assignment $\mathbf{Q}^m$:

$$\mathcal{L}^m_c = D_{KL}\left(\mathbf{P}^m \parallel \mathbf{Q}^m\right) = \sum_{i=1}^{N_m}\sum_{j=1}^{K} p^m_{ij} \log \frac{p^m_{ij}}{q^m_{ij}}. \tag{3}$$

Hence, the total loss of client m consists of two parts:

$$\mathcal{L}^m = \mathcal{L}^m_r + \gamma \mathcal{L}^m_c, \tag{4}$$

where γ is a trade-off coefficient between the clustering and reconstruction losses. The reconstruction loss $\mathcal{L}^m_r$ guarantees that the embedded features adequately represent the client's original data. The optimization of the clustering loss $\mathcal{L}^m_c$ involves extracting complementary information from other clients by minimizing the KL divergence between $\mathbf{Q}^m$ and $\mathbf{P}^m$.

3.2.2. Global data fusion training

In our architecture, the server receives embedded features and cluster assignments uploaded by each client. From the server's perspective, due to inconsistent data collection among clients, we face challenges with unaligned and incomplete data across views, rendering direct fusion for clustering impractical. To address these issues, we propose an adaptive alignment and imputation approach across clients (see Section 3.3). Then, we design a straightforward data fusion strategy on the processed data to extract global self-supervision information and enable high-quality global clustering structures.

We denote the information that has undergone adaptive alignment and adaptive imputation as $\left\{\hat{\mathbf{Z}}^m\right\}_{m=1}^M$ and $\left\{\hat{\mathbf{Q}}^m\right\}_{m=1}^M$. To extract global self-supervised information and global clustering structures from multi-view data across multiple clients, the server initially obtains global embedded features through the following equation:

$$\hat{\mathbf{Z}} = \left[\hat{\mathbf{Z}}^1, \hat{\mathbf{Z}}^2, \dots, \hat{\mathbf{Z}}^M\right] \in \mathbb{R}^{N \times \sum_{m=1}^M d_m}. \tag{5}$$

For the global prototypes $\mathbf{C} \in \mathbb{R}^{K \times \sum_{m=1}^M d_m}$, which captures the common patterns shared among samples within the same cluster, we define $\hat{\mathbf{z}}_i \in \hat{\mathbf{Z}}$ and $\mathbf{c}_j = \left[\mathbf{c}^1_j, \mathbf{c}^2_j, \dots, \mathbf{c}^M_j\right] \in \mathbf{C}$, and obtain it using the following objective:

$$\min_{\{\mathbf{c}_j\}_{j=1}^K} \sum_{i=1}^{N}\sum_{j=1}^{K} \left\|\hat{\mathbf{z}}_i - \mathbf{c}_j\right\|^2. \tag{6}$$

For the global pseudo-labels $\mathbf{P}$, which represents a unified target distribution obtained by the server, fuses information from all clients. Through self-supervised learning, this approach aligns the local model training of clients with the global objective. We quantify the similarity between global embedded features and global prototypes by transforming the Euclidean distance between them into conditional probabilities by Student's t-distribution [45]:

$$t_{ij} = \frac{\left(1 + \left\|\mathbf{z}_i - \mathbf{c}_j\right\|^2\right)^{-1}}{\sum_j \left(1 + \left\|\mathbf{z}_i - \mathbf{c}_j\right\|^2\right)^{-1}}. \tag{7}$$

To increase the discriminability of the pseudo soft assignments, the global pseudo-labels $\mathbf{P}$ are computed by:

$$p_{ij} = \frac{\left(t_{ij} / \sum_j t_{ij}\right)^2}{\sum_j \left(t_{ij} / \sum_j t_{ij}\right)^2} \in \mathbf{P}. \tag{8}$$

The global prototypes $\mathbf{C}$ and global pseudo-labels $\mathbf{P}$ form the global self-supervision information, which is distributed to each client. Furthermore, the high-quality global clustering structures are obtained by fusing the clustering assignments $\left\{\hat{\mathbf{Q}}^m\right\}_{m=1}^M$ from all clients. These clustering assignments also undergo adaptive alignment and imputation. The clustering prediction of the ith sample is calculated by

$$y_i = \arg\max_j \left(\frac{1}{M}\sum_{m=1}^{M} \hat{q}^m_{ij}\right). \tag{9}$$

3.3. Cross-client adaptive alignment and imputation

In the problem formulation section, we take client m as an example and divide its local private data into $\mathbf{X}^m = \left[\mathbf{X}^m_A; \mathbf{X}^m_U; \mathbf{X}^m_I\right]$. Following local model training and data interaction, the server obtains the corresponding view-specific embedded features $\mathbf{Z}^m = \left[\mathbf{Z}^m_A; \mathbf{Z}^m_U; \mathbf{Z}^m_I\right]$ and clustering assignments $\mathbf{Q}^m = \left[\mathbf{Q}^m_A; \mathbf{Q}^m_U; \mathbf{Q}^m_I\right]$. We initially employ the aligned and complete information $\left\{\mathbf{Z}^m_A\right\}_{m=1}^M$ to initialize the global prototypes $\mathbf{C}$ by Eq. (6), and subsequently update them in the data fusion module. At this point, if we directly concatenate the embedded features of each client for clustering, unaligned and incomplete data will significantly impact the clustering results. These insights aid in training the adaptive alignment and imputation modules, correcting the original information based on training results.

3.3.1. Adaptive alignment

To fix the unaligned information, we design the adaptive alignment module. This module conducts network training using the sample commonality features from the cross-client alignment parts. Simultaneously, it computes alignment matrices adaptively to adjust the unaligned parts across clients.

Since the independent training of local models on each client's private data without parameter sharing, there is no guarantee that the extracted embedded features belong to the same low-dimensional space. Therefore, the server employs $\{\mathbf{H}^m = \mathbf{Q}^m\mathbf{C}\}_{m=1}^M$ to capture the sample commonality features from different client perspectives, where $\mathbf{H}^m \in \mathbb{R}^{N_m \times \sum_{m=1}^M d_m}$ represents these features obtained from mth client. Then we propose to use the sample commonality features from the cross-client alignment parts $\{\mathbf{H}_A^m = \mathbf{Q}_A^m\mathbf{C}\}_{m=1}^M$ to train a neural network with the following objective function:

$$\mathcal{L}_1 = \sum_{m=1}^{M} \left\| \mathbf{H}_A^a \mathbf{W}_H^a - \mathbf{A}^m \tilde{\mathbf{H}}_A^m \mathbf{W}_H^m \right\|_F^2 + \alpha \sum_{m=1}^{M} f(\mathbf{A}^m - \mathbf{A}_{gt}^m), \tag{10}$$

where $\{\mathbf{W}_H^m\}_{m=1}^M$ represents the mined view versatility, $\{\mathbf{A}^m\}_{m=1}^M$ are the alignment matrices computed by this adaptive module, and $\left\{\mathbf{A}_{gt}^m\right\}_{m=1}^M$ are the permutation ground truths derived from $\{\tilde{\mathbf{H}}_A^m\}_{m=1}^M$ w.r.t. $\{\mathbf{H}_A^m\}_{m=1}^M$. $\mathbf{H}_A^a$ and $\mathbf{W}_H^a$ are discovered by the anchor client, which is the client with the best training results from the previous local training round. Here, $f(\cdot)$ is a specific regularizer function, and $\alpha > 0$ is a balancing parameter. As the training of the network progresses, the alignment matrices evolve dynamically. The alignment matrix for client m is computed using the following formula:

$$\mathbf{A}^m = \mathrm{softmax}\left(\frac{\left(\mathbf{H}^a \mathbf{W}_H^a\right)\left(\mathbf{H}^m \mathbf{W}_H^m\right)^T}{\sqrt{\sum_{m=1}^M d_m}} \right). \tag{11}$$

The first term of the loss function $\mathcal{L}_1$ (in Eq. (10)) measures the difference in the non-linearly transformed sample commonality features between the anchor client and other clients. This term serves a dual purpose: learning view versatility while quantifying alignment effectiveness. Additionally, the alignment matrices are used as network parameters, which are continuously tuned during training. The second term acts as a constraint, encouraging alignment matrices to close their corresponding ground truth matrices.

Then the unaligned data are adjusted through this adaptive module. Taking client m as an example, we obtain the adjusted embedded features, $\hat{\mathbf{Z}}^m = [\mathbf{Z}_A^m; \mathbf{P}^m\mathbf{Z}_U^m; \mathbf{Z}_I^m]$, and clustering assignments, $\hat{\mathbf{Q}}^m = [\mathbf{Q}_A^m; \mathbf{P}^m\mathbf{Q}_U^m; \mathbf{Q}_I^m]$.

3.3.2. Adaptive imputation

For incomplete information across clients, we propose utilizing sample commonality and view versatility for imputation and adaptive data extension based on imputation quality.

To observe the cross-client complete data from the server's perspective, we employ an indicator matrix $\mathbf{I} \in 0,1^{N \times M}$, where $\mathbf{I}_{im} = 1$ if the ith sample exists in the mth client; otherwise, $\mathbf{I}_{im} = 0$. For client m, if $\mathbf{z}_i^m \in \mathbf{Z}^m$ and $\sum_{m=1}^M \mathbf{I}_{im} = M$, then $\mathbf{z}_i^m \notin \mathbf{Z}_I^m$; otherwise, $\mathbf{z}_i^m \in \mathbf{Z}_I^m$. Using the defined indicator matrix, we obtain the global cluster assignments $\mathbf{Q}$ to aid us in adaptive imputation by the following formula:

$$q_i = \frac{\sum_{m=1}^M \mathbf{I}_{im} q_i^m}{\sum_{m=1}^M \mathbf{I}_{im}} \in \mathbf{Q}. \tag{12}$$

Based on the acquired sample commonality features $\mathbf{QC}$, we leverage the complete and aligned information across clients to learn view-specific patterns $\{\mathbf{W}_I^m\}_{m=1}^M$ and uncover view versatility, as outlined below:

$$\min_{\left\{\mathbf{W}_I^m\right\}_{m=1}^M} \sum_{\mathbf{z}_i^m \in \mathbf{Z}_A^m} \sum_{m=1}^{M} \left\| \mathbf{z}_i^m - \mathbf{W}_I^m \mathbf{q}_i \mathbf{C}^m \right\|_2^2. \tag{13}$$

Algorithm 1 Federated multi-view Clustering for Unaligned and Incomplete data Fusion (FCUIF)

Input: Data with M views $\{\mathbf{X}^m\}_{m=1}^M$ distributed across M clients, number of clusters K, parameters γ and α, communication rounds R.
Output: Global clustering predictions $\mathbf{Y} = \{y_1, y_2, \ldots, y_n\}$.

```
while not reaching R rounds do
    for m = 1 to M do in parallel
        if not initialized then
            Pretrain the autoencoders by optimizing Eq. (1).
            Initialize u^m by global prototypes C.
        else
            Optimize Eq. (4), update θ^m, φ^m, and u^m.
        end if
        Upload Z^m and Q^m to the server.
    end for
    Cross-client adaptive alignment by Eqs. (10)-(11).
    Cross-client adaptive imputation by Eqs. (13)-(14).
    Obtain global embedded features Z by Eq. (5).
    Obtain global prototypes C by Eq. (6).
    Obtain global pseudo-labels P by Eqs. (7)-(8).
    Distribute C and P to each client.
end while
Calculate the clustering predictions by Eq. (9).
```

We can utilize the extracted sample commonality features $\mathbf{QC}$ and view-specific patterns $\{\mathbf{W}_I^m\}_{m=1}^M$ to guide the imputation of incomplete embedded features $\{\mathbf{Z}_I^m\}_{m=1}^M$. In this case, when $\mathbf{I}_{im} = 0$, the incomplete embedded feature $\mathbf{z}_i^m$ can be imputed as follows:

$$\mathbf{z}_i^m = \mathbf{W}_I^m \mathbf{q}_i \mathbf{C}^m \in \mathbf{Z}_I^m. \tag{14}$$

As the imputation process progresses, the data expand, providing access to more information. Higher-quality imputed values can lead to more comprehensive and accurate global embedded features during the data fusion process. However, a higher missing rate can limit clients' representation of view characteristics, leading to imprecise imputation. Additionally, the imbalanced distribution of data across clients can bias the view information for certain clients. To address these limitations, we evaluate the global embedded features obtained after imputation. If the clustering results of the imputed global embedded features are inferior to those without imputation, we abandon imputation and proceed with training using the non-imputed global embedded features.

The information, which has undergone adaptive alignment and adaptive imputation, including embedded features $\left\{\hat{\mathbf{Z}}^m\right\}_{m=1}^M$ and clustering assignments $\left\{\hat{\mathbf{Q}}^m\right\}_{m=1}^M$, is passed into the data fusion module. This facilitates the discovery of more accurate global self-supervision information, making it easier to achieve high-quality global clustering.

3.4. Optimization

The optimization of FCUIF is outlined in Algorithm 1, consisting of two main parts: the clients and the server. Clients are responsible for parallel training local models. In the initial round, they pretrain autoencoder $f_{\theta^m}^m$ and $g_{\phi^m}^m$ by Eq. (1). In subsequent rounds, they utilize the global self-supervision information provided by the server to conduct view-specific local training in Eq. (4). The server receives embedded features and cluster assignments uploaded by each client and extracts sample commonality and view versatility. For unaligned information across clients, adaptive alignment is performed by Eqs. (10) and (11). For incomplete information across clients, adaptive data extension is performed based on imputation quality by Eqs. (13) and (14). Subsequently, the server leverages the information post-adaptive alignment and imputation to construct global embedded features and discover global self-supervision information, which includes global prototypes and global pseudo-labels. Finally, we fuse the clustering assignments from all clients to generate the global clustering predictions by Eq. (9).

4. Experiments

4.1. Experimental settings

4.1.1. Datasets

We conduct the experiments on four commonly-used datasets. Concretely, **BDGP** [46] contains 2500 samples in five categories, each sample with two views, including texture and visual features. **Reuters** [47] includes 1200 articles across six categories, each article available in five languages as five text views. **Scene** [48] consists of 4485 scene images categorized into 15 classes, each represented by three views. **Handwritten Numerals (HW)**[1] comprises 2000 samples in ten numeral categories, each with six visual views derived from binary images.

Note that in our federated setting, multiple views of these datasets are distributed among different clients and are isolated from each other. We primarily simulate three scenarios. (1) Only unaligned data are present. To simulate unaligned multi-view data, each client internally shuffles its data randomly, following [31]. (2) Only incomplete data are present. For simulating incomplete multi-view data, we randomly remove samples from various views, ensuring that each sample retains one view, as outlined in [49]. (3) Unaligned and incomplete data coexist. To simulate unaligned and incomplete multi-view data, we intentionally omit data and then randomly shuffle the complete parts of the incomplete multi-view data within each client. Additionally, we define two important parameters: the unaligned rate $\delta_1 = m_1/n_c$, where m_1 is the count of shuffled samples, and the missing rate $\delta_2 = m_2/n$, where m_2 represents the number of samples with incomplete information across all clients. Here, n_c is the count of samples with complete data across all clients, expressed as $n(1-\delta_2)$, with n denoting the overall size of datasets.

4.1.2. Comparing methods

We select several relevant centralized algorithms and our previous work as comparison methods. For the issue of unaligned data across clients, akin to the partially view-aligned problem in MVC, we compare our method against five additional state-of-the-art methods:

- PVC (2020) [31] establishes category-level correspondences for handling unaligned multi-view data in the latent space learned.
- MvCLN (2021) [50] learns aligned data at the category level using a noise-resistant contrastive loss for handling unaligned multi-view data.
- GWMAC (2022) [51] utilizes the Gromov-Wasserstein barycenter to achieve data alignment and clustering for multi-view data.
- SURE (2022) [32] addresses partially view-unaligned and partially sample-missing problems within a unified framework.
- SMILE (2023) [52] learns consensus semantics for realigning/imputing defective instances and forming clusters.

The issue of incomplete data across clients parallels the problem tackled by incomplete multi-view clustering. We compare our method against eight state-of-the-art methods in the field:

- GIMC-FLSD (2020) [53] considers the local geometric information and the unbalanced discriminating powers of IMVC.
- DCP (2021) [17] is a deep IMVC method that leverages a contrastive prediction module to recover the missing data.
- HCP-IMSC (2022) [54] is an IMVC method that preserves sample and views high-order correlations.
- IMVC-CBG (2022) [55] leverages anchor learning and consensus bipartite graph for handling large-scale IMVC.
- DSIMVC (2022) [56] reduces the risk of clustering performance degradation caused by the semantic inconsistency in estimated views in IMVC, both theoretically and experimentally.
- LSIMVC (2022) [57] learns a sparse and structured consensus latent representation from incomplete multi-view data by optimizing a graph-embedded multi-view matrix factorization model.
- PGP (2023) [58] is a graph based IMVC method where the associated missing entries can be inferred through graph propagation.
- FedDMVC (2023) [14] is a federated multi-view clustering method that simultaneously addresses feature heterogeneity and IMVC.

4.1.3. Implementation details

Our method is implemented using PyTorch and the Flower federated learning framework [59]. We use the same fully connected (Fc) autoencoder structure for all four datasets, following [44]. The encoder structure for each client is Input- $\mathrm{Fc}_{500}-\mathrm{Fc}_{500}-\mathrm{Fc}_{2000}-\mathrm{Fc}_{10}$, and the decoder is symmetric with the encoder. All the autoencoders are pretrained for 500 epochs and the dimensionality of all clients' embedded features is reduced to 10. Also, we set the trade-off coefficient $\gamma = 0.1$ and $\alpha = 0.1$, use the batch size of 256, and set the number of local epochs to 300. Additionally, two three-layer MLPs with architectures $d_m-32-64-d_m$ are employed on the server to extract view versatility, auxiliary adaptive alignment, and adaptive imputation, respectively. ReLU is the activation function of all hidden layers and Adam (default learning rate is 0.001) is chosen as the optimizer. We set communication rounds R to 3, which means that the server and each client communicate for 3 rounds.

4.1.4. Evaluation measures

We evaluate the clustering performance using three widely recognized metrics: accuracy (ACC), normalized mutual information (NMI), and adjusted rand index (ARI). Higher values for these metrics indicate improved clustering results.

4.2. Clustering results

4.2.1. Cross-client partially unaligned

In this scenario, we examine the impact of unaligned rates ranging from 0.1 to 0.7 with an interval of 0.2 on the clustering performance of four datasets, as shown in Fig. 2. Compared to existing MVC methods for partially view-aligned problems, FCUIF consistently outperforms them across all four datasets, affirming the effectiveness of our adaptive alignment module. This module adaptively computes alignment matrices, facilitating high-quality cross-client alignment.

4.2.2. Cross-client partially missing

In this scenario, we investigate how missing rates ranging from 0.1 to 0.7 with an interval of 0.2 affect the clustering performance of four datasets, as shown in Fig. 3. Compared with other IMVC methods, even with a relatively high missing rate, FCUIF can learn a consensus representation that guides the imputation of missing samples from each client, thus avoiding the negative impact of low-quality samples.

4.2.3. Cross-client partially unaligned and partially missing

We conduct comparative analyses for three scenarios: partially unaligned ($\delta_1 = 0.5$), partially missing ($\delta_2 = 0.5$), and both partially unaligned and missing ($\delta_1 = 0.5$ and $\delta_2 = 0.5$). Experimental results are presented in Table 1. These three scenarios are denoted as types (a), (b), and (c). Given the limited availability of MVC methods capable of simultaneously addressing unaligned and missing data, we only compare FCUIF with the SURE method in type (c). The results demonstrate FCUIF's adaptability to various scenarios involving unaligned and missing data. This adaptability can be attributed to the excellent performance of our adaptive alignment and imputation modules, which

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

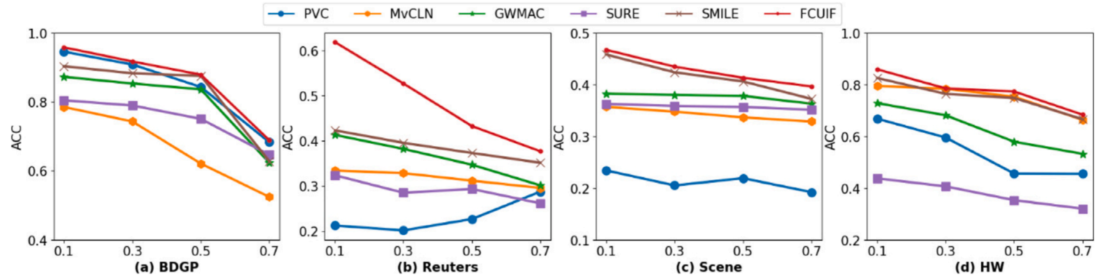


Fig. 2. Performance analysis on four datasets with different unaligned rates.

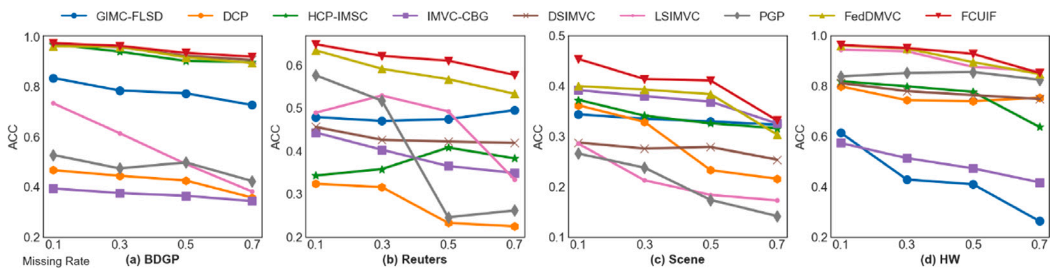


Fig. 3. Performance analysis on four datasets with different missing rates.

Table 1
Experiments results on BDGP and Reuters. The best result in each column is shown in bold. Here, (a) denotes partially unaligned, (b) denotes partially missing, and (c) denotes partially unaligned and partially missing.

Type	Methods	BDGP			Reuters			Scene			HW		
		ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
(a)	PVC [31]	0.843	0.630	0.647	0.227	0.093	0.023	0.219	0.191	0.073	0.456	0.470	0.311
	MvCLN [50]	0.621	0.437	0.373	0.312	0.113	0.051	0.337	0.308	0.175	0.753	0.659	0.515
	GWMAC [51]	0.836	0.627	0.671	0.347	0.135	0.128	0.378	0.322	0.227	0.579	0.580	0.428
	SURE [32]	0.750	0.502	0.461	0.293	0.091	0.046	0.357	0.307	0.186	0.353	0.300	0.175
	SMILE [52]	0.875	0.698	0.714	0.373	0.165	0.123	0.406	0.367	0.234	0.748	0.645	0.579
	FCUIF (ours)	**0.878**	**0.703**	**0.726**	**0.433**	**0.232**	**0.198**	**0.413**	**0.377**	**0.240**	**0.773**	**0.666**	**0.586**
(b)	GIMC-FLSD [53]	0.772	0.532	0.534	0.473	0.275	0.202	0.300	0.264	0.135	0.408	0.431	0.229
	DCP [17]	0.424	0.305	0.054	0.232	0.137	0.013	0.328	0.346	0.214	0.738	0.734	0.626
	HCP-IMSC [54]	0.901	0.769	0.759	0.407	0.219	0.136	0.325	0.273	0.143	0.775	0.710	0.651
	IMVC-CBG [55]	0.363	0.176	0.056	0.364	0.213	0.088	0.268	0.270	0.144	0.471	0.473	0.237
	DSIMVC [56]	0.921	0.829	0.834	0.421	0.256	0.187	0.278	0.304	0.145	0.762	0.736	0.650
	LSIMVC [57]	0.490	0.388	0.303	0.152	0.062	0.199	0.212	0.218	0.092	0.874	0.828	0.845
	PGP [58]	0.496	0.310	0.398	0.245	0.088	0.285	0.237	0.268	0.139	0.854	0.843	0.842
	FedDMVC [14]	0.915	0.774	0.803	0.566	0.299	0.249	0.393	0.343	0.225	0.893	0.824	0.790
	FCUIF (ours)	**0.933**	**0.862**	**0.859**	**0.609**	**0.407**	**0.357**	**0.410**	**0.376**	**0.229**	**0.926**	**0.844**	**0.856**
(c)	PVC [31]	0.468	0.267	0.073	0.249	0.113	0.013	0.248	0.278	0.071	0.393	0.352	0.294
	MvCLN [50]	0.572	0.380	0.365	0.253	0.136	0.144	0.266	0.185	0.134	0.475	0.366	0.239
	GWMAC [51]	0.751	0.592	0.560	0.301	0.078	0.083	0.326	0.313	0.173	0.482	0.453	0.361
	SURE [32]	0.734	0.497	0.462	0.286	0.066	0.038	0.342	0.262	0.165	0.309	0.209	0.111
	SMILE [52]	0.826	0.622	0.615	0.397	0.172	0.126	0.348	0.347	0.209	0.773	0.732	0.665
	FCUIF (ours)	**0.835**	**0.630**	**0.639**	**0.508**	**0.258**	**0.194**	**0.354**	**0.355**	**0.214**	**0.785**	**0.738**	**0.688**

effectively avoid the adverse effects of unaligned and incomplete parts of the data on clustering.

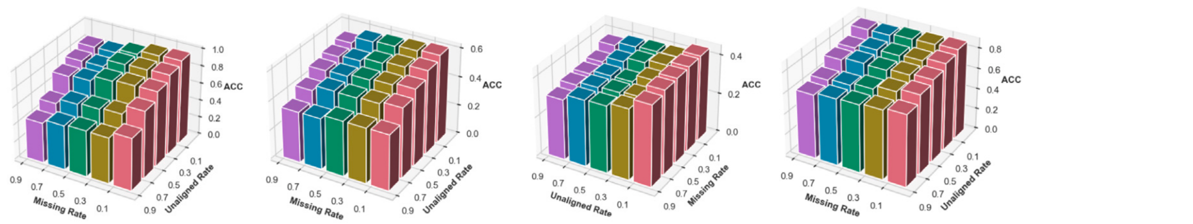
In our experimental setup, unaligned exists only for complete parts. Therefore, even though types (a) and (c) run under the same unaligned rate, (c) has fewer unaligned samples than (a) due to the presence of missing samples. It is worth noting that on Reuters and HW datasets, our method yields better clustering results for (c) compared to (a) under the same unaligned rate. These results reflect that these two datasets are more affected by unaligned data compared to missing data due to the higher number of views.

Moreover, we conduct experiments on four datasets by varying both the unaligned rate and missing rate from 0.1 to 0.9 with intervals of 0.2, as shown in Fig. 4. With increasing unaligned and incomplete rates, FCUIF's performance declines. However, on the Scene and HW datasets, the decrease is relatively small, indicating strong robustness. On the BDGP and Reuters datasets, ACC shows minimal variation within a certain range ($\delta_1 \leqslant 0.5$ and $\delta_2 \leqslant 0.5$), demonstrating a degree of robustness. Experimental results show that FCUIF effectively adapts to varying levels of unaligned and incomplete data, enabling high-quality data fusion and cluster structure discovery.

4.3. Model analysis

4.3.1. Ablation study

Components A and B respectively represent the constraint terms in the adaptive alignment module and the view-specific patterns $\mathbf{W}_H$ mined during its training. Components C and D respectively represent the view-specific patterns $\mathbf{W}_I$ discovered during the training of the adaptive imputation module and the utilization of adaptive imputation strategies. As shown in Table 2, it is found that the performance of clustering is significantly influenced by component D. We can also find that combining components B and D yields better results than using

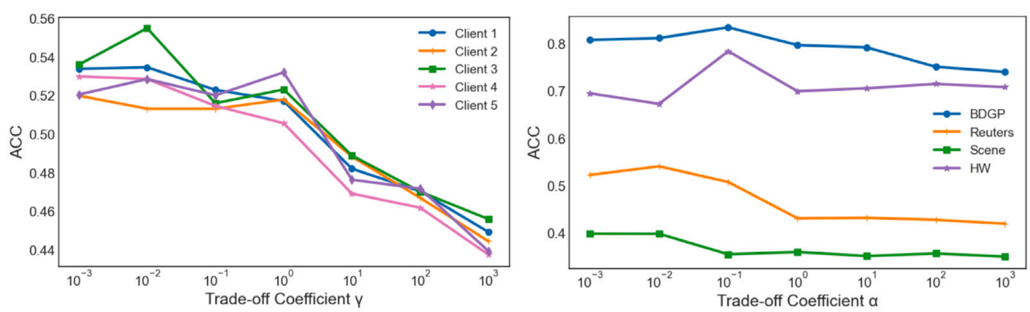


(a) BDGP (b) Reuters (c) Scene (d) HW

Fig. 4. ACC on four datasets with different unaligned rates and missing rates.

Table 2
Ablation studies on four datasets when $\delta_1 = 0.5$ and $\delta_2 = 0.5$.

	Components				BDGP			Reuters			Scene			HW		
	A	B	C	D	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
Item-1		✓	✓	✓	0.800	0.582	0.509	0.502	0.221	0.160	0.343	0.323	0.205	0.754	0.708	0.650
Item-2	✓		✓	✓	0.821	0.614	0.617	0.466	0.246	0.147	0.325	0.292	0.157	0.701	0.657	0.576
Item-3	✓	✓		✓	0.791	0.559	0.561	0.432	0.235	0.154	0.328	0.308	0.162	0.741	0.734	0.655
Item-4	✓	✓	✓		0.756	0.525	0.513	0.452	0.235	0.149	0.317	0.285	0.169	0.729	0.708	0.632
Item-5	✓	✓	✓	✓	0.835	0.630	0.639	0.508	0.258	0.194	0.354	0.355	0.214	0.785	0.738	0.688



(a) ACC with different γ on Reuters (b) ACC with different α on four datasets

Fig. 5. ACC w.r.t. different parameter settings when $\delta_1 = 0.5$ and $\delta_2 = 0.5$.

either B or D alone, which illustrates that view-specific patterns mined and adaptive imputation strategies play a crucial role in the overall performance of the model. With the high confidence imputation, component A is also indispensable, which can improve consistency between different views that is beneficial for clustering.

4.3.2. Parameter analysis

In our study, the FCUIF has two primary hyper-parameters, namely α and γ. To showcase the stability of the FCUIF method, we conduct experiments with various parameter settings and evaluate the resulting clustering performance, as depicted in Fig. 5. We observe that higher values of γ promote disentanglement but introduce a trade-off between the fidelity of reconstructions and the disentanglement of latent features. Moreover, excessively large or small values of γ adversely affect clustering performance by creating an imbalance between reducing redundant information and obtaining consistent information. The effect of parameter α is relatively weak, so we choose the middle value to ensure the quality of the alignment matrices. Based on our experimental findings, we recommend setting both α and γ to 0.1 for optimal performance.

4.3.3. Missing strategies

We investigate the performance of FCUIF under various missing strategies on four datasets when $\delta_1 = 0.5$ and $\delta_2 = 0.5$, as shown in Fig. 6. Point-wise missing refers to scattered data missing, commonly employed in many IMVC methods. Block-wise missing occurs when data are missing in the form of blocks, such as when monitoring devices may fail over several hours, leading to continuous data missing. For block-wise missing, some IMVC methods that merge adjacent information may result in unavailability. In this experiment, we define missing involving more than five consecutive samples as block-wise missing. Imbalanced missing, a unique challenge introduced by federated learning, implies potential imbalances in the number of samples across clients. We utilize the Dirichlet distribution to simulate cross-client imbalanced missing.

The results demonstrate FCUIF's adaptability to various missing strategies. Furthermore, performance tends to degrade with block-wise missing compared to point-wise missing, underscoring the importance of addressing block-wise missing. Imbalanced missing has a positive impact on BDGP but yields adverse effects on the other three datasets. This arises from BDGP having only two views with significant quality disparities. Imbalanced missing prompts FCUIF to focus more on the higher-quality view, resulting in improved performance. Conversely, for the other three datasets where view quality differences are less pronounced, imbalanced missing leads to information loss and negative effects.

5. Conclusion

In this paper, we present FCUIF, a novel federated multi-view clustering method designed to address the challenges of unaligned and incomplete data across distributed clients in multi-view datasets, thereby facilitating effective data fusion. For unaligned data, FCUIF adaptively computes alignment matrices based on sample commonality and view versatility, achieving cross-client alignment. For incomplete data, FCUIF assesses imputation quality in an unsupervised

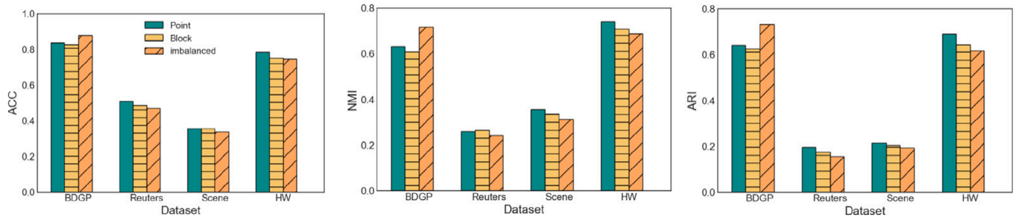


Fig. 6. Performance analysis on four datasets with different missing strategies.

manner, enabling adaptive imputation. Furthermore, FCUIF effectively addresses various scenarios of incomplete multi-view data, including point, block, and imbalanced missing strategies. To enhance clustering, FCUIF also designs a data fusion strategy on the server to extract high-quality global clustering structures. Extensive experiments on four public datasets demonstrate FCUIF's superior performance in handling unaligned and incomplete multi-view data.

CRediT authorship contribution statement

Yazhou Ren: Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Software, Supervision, Validation, Writing – original draft, Writing – review & editing. **Xinyue Chen:** Conceptualization, Formal analysis, Investigation, Methodology, Software, Validation, Writing – original draft, Writing – review & editing. **Jie Xu:** Formal analysis, Investigation, Methodology, Resources, Validation, Writing – original draft, Writing – review & editing. **Jingyu Pu:** Formal analysis, Investigation, Methodology, Resources, Validation, Writing – original draft, Writing – review & editing. **Yonghao Huang:** Formal analysis, Investigation, Methodology, Resources, Validation, Writing – original draft, Writing – review & editing. **Xiaorong Pu:** Formal analysis, Investigation, Methodology, Resources, Validation, Writing – original draft, Writing – review & editing. **Ce Zhu:** Formal analysis, Investigation, Methodology, Resources, Validation, Writing – original draft, Writing – review & editing. **Xiaofeng Zhu:** Formal analysis, Investigation, Methodology, Resources, Validation, Writing – original draft, Writing – review & editing. **Zhifeng Hao:** Formal analysis, Investigation, Methodology, Resources, Validation, Writing – original draft, Writing – review & editing. **Lifang He:** Conceptualization, Formal analysis, Investigation, Methodology, Project administration, Resources, Supervision, Validation, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work was supported in part by the National Key Research and Development Program of China under Grant 2020YFC2004300 and Grant 2020YFC2004302, in part by the Shenzhen Science and Technology Program under Grant JCYJ20230807115959041 and Grant JCYJ20230807120010021, in part by the Sichuan Provincial Key Laboratory of Philosophy and Social Science for Language Intelligence in Special Education under Grant YYZN-2023-3, in part by the National National Natural Science Foundation of China under Grant 61971052. Lifang He is partially supported by the NSF grants (MRI-2215789, IIS-1909879, IIS-2319451), NIH grant under R21EY034179, and Lehigh's grants under Accelerator and CORE.

References

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, B.A. y Arcas, Communication-efficient learning of deep networks from decentralized data, in: AISTATS, 2017, pp. 1273–1282.
[2] T. Li, A.K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, V. Smith, Federated optimization in heterogeneous networks, in: MLSys, Vol. 2, 2020, pp. 429–450.
[3] P. Kairouz, H.B. McMahan, B. Avent, A. Bellet, M. Bennis, A.N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al., Advances and open problems in federated learning, Found. Trends Mach. Learn. 14 (1–2) (2021) 1–210.
[4] Z. Qu, Y. Tang, G. Muhammad, P. Tiwari, Privacy protection in intelligent vehicle networking: A novel federated learning algorithm based on information fusion, Inf. Fusion 98 (2023) 101824.
[5] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, V. Chandra, Federated learning with non-iid data, 2018, arXiv preprint arXiv:1806.00582.
[6] Q. Yang, Y. Liu, T. Chen, Y. Tong, Federated machine learning: Concept and applications, ACM Trans. Intell. Syst. Technol. 10 (2) (2019) 1–19.
[7] S.K. Singh, L.T. Yang, J.H. Park, FusionFedBlock: Fusion of blockchain and federated learning to preserve privacy in industry 5.0, Inf. Fusion 90 (2023) 233–240.
[8] F. Liu, X. Wu, S. Ge, W. Fan, Y. Zou, Federated learning for vision-and-language grounding problems, in: AAAI, 2020, pp. 11572–11579.
[9] B. Xiong, X. Yang, F. Qi, C. Xu, A unified framework for multi-modal federated learning, Neurocomputing 480 (2022) 110–118.
[10] J. Chen, A. Zhang, FedMSplit: Correlation-adaptive federated multi-task learning across multimodal split networks, in: SIGKDD, 2022, pp. 87–96.
[11] Y.-M. Lin, Y. Gao, M.-G. Gong, S.-J. Zhang, Y.-Q. Zhang, Z.-Y. Li, Federated learning on multimodal data: A comprehensive survey, Mach. Intell. Res. (2023) 1–15.
[12] M. Huang, H. Li, B. Bai, C. Wang, K. Bai, F. Wang, A federated multi-view deep learning framework for privacy-preserving recommendations, 2020, arXiv preprint arXiv:2008.10808.
[13] S. Che, Z. Kong, H. Peng, L. Sun, A. Leow, Y. Chen, L. He, Federated multi-view learning for private medical data integration and analysis, ACM Trans. Intell. Syst. Technol. 13 (4) (2022) 1–23.
[14] X. Chen, J. Xu, Y. Ren, X. Pu, C. Zhu, X. Zhu, Z. Hao, L. He, Federated deep multi-view clustering with global self-supervision, in: ACM MM, 2023, pp. 3498–3506.
[15] S. Huang, W. Shi, Z. Xu, I.W. Tsang, J. Lv, Efficient federated multi-view learning, Pattern Recognit. 131 (2022) 108817.
[16] J. Guo, J. Ye, Anchors bring ease: An embarrassingly simple approach to partial multi-view clustering, in: AAAI, 2019, pp. 118–125.
[17] Y. Lin, Y. Gou, Z. Liu, B. Li, J. Lv, X. Peng, Completer: Incomplete multi-view clustering via contrastive prediction, in: CVPR, 2021, pp. 11174–11183.
[18] J. Xu, C. Li, L. Peng, Y. Ren, X. Shi, H.T. Shen, X. Zhu, Adaptive feature projection with distribution alignment for deep incomplete multi-view clustering, IEEE Trans. Image Process. 32 (2023) 1354–1366.
[19] M.-S. Chen, L. Huang, C.-D. Wang, D. Huang, Multi-view clustering in latent embedding space, in: AAAI, 2020, pp. 3513–3520.
[20] R. Li, C. Zhang, H. Fu, X. Peng, T. Zhou, Q. Hu, Reciprocal multi-layer subspace learning for multi-view clustering, in: CVPR, 2019, pp. 8172–8180.
[21] Z. Hu, F. Nie, R. Wang, X. Li, Multi-view spectral clustering via integrating nonnegative embedding and spectral embedding, Inf. Fusion 55 (2020) 251–259.
[22] H. Zhao, Z. Ding, Y. Fu, Multi-view clustering via deep matrix factorization, in: AAAI, 2017.
[23] Z. Zhang, L. Liu, F. Shen, H.T. Shen, L. Shao, Binary multi-view clustering, IEEE Trans. Pattern Anal. Mach. Intell. 41 (7) (2018) 1774–1782.
[24] Y. Ren, S. Huang, P. Zhao, M. Han, Z. Xu, Self-paced and auto-weighted multi-view clustering, Neurocomputing 383 (2020) 248–256.
[25] T. Zhou, C. Zhang, X. Peng, H. Bhaskar, J. Yang, Dual shared-specific multiview subspace clustering, IEEE Trans. Cybern. 50 (8) (2019) 3517–3530.
[26] M. Yin, W. Huang, J. Gao, Shared generative latent representation learning for multi-view clustering, in: AAAI, 2020, pp. 6688–6695.
[27] M. Hu, S. Chen, Doubly aligned incomplete multi-view clustering, 2019, arXiv preprint arXiv:1903.02785.

[28] X. Liu, M. Li, C. Tang, J. Xia, J. Xiong, L. Liu, M. Kloft, E. Zhu, Efficient and effective regularized incomplete multi-view clustering, IEEE Trans. Pattern Anal. Mach. Intell. 43 (8) (2020) 2634–2646.
[29] S.-Y. Li, Y. Jiang, Z.-H. Zhou, Partial multi-view clustering, in: AAAI, Vol. 28, 2014.
[30] H. Zhao, H. Liu, Y. Fu, Incomplete multi-modal visual data grouping, in: IJCAI, 2016, pp. 2392–2398.
[31] Z. Huang, P. Hu, J.T. Zhou, J. Lv, X. Peng, Partially view-aligned clustering, in: NeurIPS, 2020, pp. 2892–2902.
[32] M. Yang, Y. Li, P. Hu, J. Bai, J. Lv, X. Peng, Robust multi-view clustering with incomplete information, IEEE Trans. Pattern Anal. Mach. Intell. 45 (1) (2022) 1055–1069.
[33] X. Cai, F. Nie, H. Huang, Multi-view k-means clustering on big data, in: IJCAI, 2013.
[34] S. Huang, Y. Ren, Z. Xu, Robust multi-view data clustering with multi-view capped-norm k-means, Neurocomputing 311 (2018) 197–208.
[35] A. Flanagan, W. Oyomno, A. Grigorievskiy, K.E. Tan, S.A. Khan, M. Ammad-Ud-Din, Federated multi-view matrix factorization for personalized recommendations, in: ECML-PKDD, Springer, 2021, pp. 324–347.
[36] X. Xu, H. Peng, L. Sun, M.Z.A. Bhuiyan, L. Liu, L. He, Fedmood: Federated learning on mobile health data for mood detection, 2021, arXiv preprint arXiv:2102.09342.
[37] C.-M. Feng, B. Li, X. Xu, Y. Liu, H. Fu, W. Zuo, Learning federated visual prompt in null space for MRI reconstruction, in: CVPR, 2023, pp. 8064–8073.
[38] C.-M. Feng, K. Yu, N. Liu, X. Xu, S. Khan, W. Zuo, Towards instance-adaptive inference for federated learning, in: ICCV, 2023, pp. 23287–23296.
[39] K. Cheng, T. Fan, Y. Jin, Y. Liu, T. Chen, D. Papadopoulos, Q. Yang, Secureboost: A lossless federated learning framework, IEEE Intell. Syst. 36 (6) (2021) 87–98.
[40] B. Gu, A. Xu, Z. Huo, C. Deng, H. Huang, Privacy-preserving asynchronous vertical federated learning algorithms for multiparty collaborative learning, IEEE Trans. Neural Netw. Learn. Syst. 33 (11) (2021) 6103–6115.
[41] X. Peng, Z. Huang, J. Lv, H. Zhu, J.T. Zhou, COMIC: Multi-view clustering without parameter selection, in: ICML, 2019, pp. 5092–5101.
[42] J. Xu, Y. Ren, H. Tang, Z. Yang, L. Pan, Y. Yang, X. Pu, P. Yu, L. He, Self-supervised discriminative feature learning for deep multi-view clustering, IEEE Trans. Knowl. Data Eng. (2023) 7470–7482.
[43] J. Xie, R. Girshick, A. Farhadi, Unsupervised deep embedding for clustering analysis, in: ICML, 2016, pp. 478–487.
[44] X. Guo, L. Gao, X. Liu, J. Yin, Improved deep embedded clustering with local structure preservation, in: IJCAI, 2017, pp. 1753–1759.
[45] L. Van der Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (11) (2008).
[46] X. Cai, H. Wang, H. Huang, C. Ding, Joint stage recognition and anatomical annotation of drosophila gene expression patterns, Bioinformatics 28 (12) (2012) i16–i24.
[47] M.R. Amini, N. Usunier, C. Goutte, Learning from multiple partially observed views-an application to multilingual text categorization, in: NeurIPS, 2009, pp. 1–9.
[48] L. Fei-Fei, P. Perona, A bayesian hierarchical model for learning natural scene categories, in: CVPR, 2005, pp. 524–531.
[49] J. Xu, C. Li, Y. Ren, L. Peng, Y. Mo, X. Shi, X. Zhu, Deep incomplete multi-view clustering via mining cluster complementarity, in: AAAI, 2022, pp. 8761–8769.
[50] M. Yang, Y. Li, Z. Huang, Z. Liu, P. Hu, X. Peng, Partially view-aligned representation learning with noise-robust contrastive loss, in: CVPR, 2021, pp. 1134–1143.
[51] F. Gong, Y. Nie, H. Xu, Gromov–Wasserstein multi-modal alignment and clustering, in: CIKM, 2022, pp. 603–613.
[52] P. Zeng, M. Yang, Y. Lu, C. Zhang, P. Hu, X. Peng, Semantic invariant multi-view clustering with fully incomplete information, IEEE Trans. Pattern Anal. Mach. Intell. (2023).
[53] J. Wen, Z. Zhang, Z. Zhang, L. Fei, M. Wang, Generalized incomplete multi-view clustering with flexible locality structure diffusion, IEEE Trans. Cybern. (2020).
[54] Z. Li, C. Tang, X. Zheng, X. Liu, W. Zhang, E. Zhu, High-order correlation preserved incomplete multi-view subspace clustering, IEEE Trans. Image Process. 31 (2022) 2067–2080.
[55] S. Wang, X. Liu, L. Liu, W. Tu, X. Zhu, J. Liu, S. Zhou, E. Zhu, Highly-efficient incomplete large-scale multi-view clustering with consensus bipartite graph, in: CVPR, 2022, pp. 9776–9785.
[56] H. Tang, Y. Liu, Deep safe incomplete multi-view clustering: Theorem and algorithm, in: ICML, 2022, pp. 21090–21110.
[57] C. Liu, Z. Wu, J. Wen, Y. Xu, C. Huang, Localized sparse incomplete multi-view clustering, IEEE Trans. Multimed. (2022).
[58] C. Liu, R. Li, S. Wu, H. Che, D. Jiang, Z. Yu, H.-S. Wong, Self-guided partial graph propagation for incomplete multi-view clustering, IEEE Trans. Neural Netw. Learn. Syst. (2023).
[59] D.J. Beutel, T. Topal, A. Mathur, X. Qiu, J. Fernandez-Marques, Y. Gao, L. Sani, K.H. Li, T. Parcollet, P.P.B. de Gusmão, et al., Flower: A friendly federated learning research framework, 2020, arXiv preprint arXiv:2007.14390.