Check for updates

Federated semi-supervised learning based on feature alignment and knowledge distillation

Zhe Ding[1,2], Hao Yi[3,4]*, Wenrui Xie[3,4], Yuxuan Xiao[1],
Qixu Wang[1,2], Qing Chen[5], Zhiguang Qin[1] and Dajiang Chen[1]

[1]School of Information and Software Engineering, University of Electronic Science and Technology of
China, Chengdu, China, [2]School of Cybersecurity, Chengdu University of Information Technology,
Chengdu, China, [3]China Electronic Products Reliability and Environmental Testing Research Institute,
Guangzhou, China, [4]Key Laboratory of the Ministry of Industry and Information Technology for
Performance and Reliability Evaluation of Software and Hardware for Information Technology
Application Innovation Foundation, Guangzhou, China, [5]Accelink technologies Co., LTD., Wuhan, China

Introduction: Recently, federated learning has been successfully applied in fields related to cyber-physical-social systems (CPSSs), owing to its ability to harness decentralized clients for training a global model while ensuring data privacy. The existing methods encounter two main obstacles, namely, the statistical distribution heterogeneity [non-independent and identically distributed (non-IID)] among clients and the scarcity of labeled data.

Methods: In this article, we propose a federated semi-supervised learning (FSSL) model under the label-at-server scenario, denoted as FedAlign, which is tailored for distributed cyber-physical-social systems. FedAlign adopts a dual knowledge distillation framework to train the global model. On the client side, FedAlign integrates contrastive learning, knowledge distillation, and pseudo-labeling technology to train local models. The goal is to ensure that global knowledge is not overlooked while enabling clients to learn local knowledge. Meanwhile, on the server side, FedAlign utilizes maximum mean discrepancy to generate a global feature space. Based on the generated feature space, FedAlign employs a knowledge distillation mechanism and supervised learning to aggregate local knowledge and update the global model.

Results: Two classic datasets, CIFAR-10 and Fashion-MNIST, are used to evaluate the performance of FedAlign. The experimental results demonstrate that FedAlign outperforms traditional federated semi-supervised learning models.

Discussion: The integration of feature alignment and knowledge enables balancing local knowledge learning and aggregation of global model. As a consequence, FedAlign enhances the adaptability and generalization ability of the global model in CPSSs.

KEYWORDS

cyber-physical-social systems, feature space alignment, federated semi-supervised learning, knowledge distillation, maximum mean discrepancy

1 Introduction

In recent years, cyber-physical-social systems (CPSSs) have received more attention in the academic community [1–4]. Due to the characteristics of CPSSs, the common CPSS

is built on a distributed network [5, 6]. Existing deep learning models for CPSSs face two challenges, namely, data silos and the scarcity of labeled data [7]. A complete CPSS involves multiple systems or institutions that operate independently. According to some local laws, CPSSs can never gather raw data unconditionally into a central server. This has formed various data silos [8]. A large amount of data is stored in distributed CPSS terminals. Most of the data are unlabeled [9]. Thus, traditional deep learning based on centralized training cannot be directly applied to the distributed CPSS environment [10].

Federated semi-supervised learning (FSSL) combines semi-supervised learning (SSL) with federated learning (FL) to enable multiple independent clients to collaborate and effectively train a global model under the constraint of scarce labeled data without sharing raw data in the distributed CPSS environment [11]. Thus, FSSL has emerged as an efficient tool for a CPSS to train a global model in the distributed network environment [12–15]. According to the location of labeled data, FSSL scenarios can be categorized into the label-at-server and label-at-client. In the label-at-server scenario, the data on the client side consist solely of unlabeled data, while the server holds labeled data. By contrast, in the label-at-client scenario, the data on the client side include both labeled and unlabeled data [16, 17]. From the perspective of model training, the label-at-server scenario poses greater challenges. Moreover, the label-at-server scenario is more prevalent than the label-at-client scenario [18]. Thus, this article focuses on label-at-server-based FSSL.

The framework of FSSL under the labeled-at-server scenario is shown in Figure 1. For the label-at-server scenario, clients train local models using self-supervised learning on their unlabeled data, while the server utilizes local models uploaded by the clients to generate a global model and uses its labeled data on the server to optimize the global model. Existing FSSL under the labeled-at-server scenario faces major challenges as follows: [19–22].

- **The statistical distribution heterogeneity (non-IID) across clients:** Within the framework of FSSL, data inherently exhibit the non-IID characteristic among a large number of participating clients. This arises because each client, such as mobile phones and Internet of Things (IoT) sensors, generates and stores data based on its unique local environment and user-specific usage patterns. Consequently, the local data on a given client are not a representative sample of the global distribution but rather a biased reflection of its individual experiences. Furthermore, the non-IID characteristic can lead to issues such as difficulties in global model convergence, global model bias, and client drift, thereby reducing the training efficiency of the FL model.
- **Difference for optimization goal:** For the label-at-server scenario, the server's optimization goal is to make the global model align with its small amount of labeled data—that is, the objective corresponds to supervised learning. The client's optimization goal is to make the local model align with the distribution of local unlabeled data—that is, the objective corresponds to unsupervised learning. These two goals may be completely misaligned.

In this article, a novel FSSL model based on the label-at-server scenario for a distributed CPSS, denoted as FedAlign, is proposed

to address the aforementioned obstacles. FedAlign leverages the teacher–student framework on both the client side and server side. The teacher model serves as a repository of global knowledge, and the student model is responsible for learning local knowledge. For the first obstacle, FedAlign applies maximum mean discrepancy (MMD) to achieve feature space alignment between the client and server sides to optimize the global model. Furthermore, FedAlign applies a knowledge distillation mechanism to infuse global knowledge into the client-side local training. For the second obstacle, FedAlign integrates contrastive learning, a knowledge distillation mechanism, and pseudo-label-based supervised learning to improve the effectiveness of local training on the client side. Contrastive learning is used to improve the model's representational capacity for unlabeled data, while the knowledge distillation mechanism is applied to infuse global knowledge into local training. The main contributions of this article are outlined below.
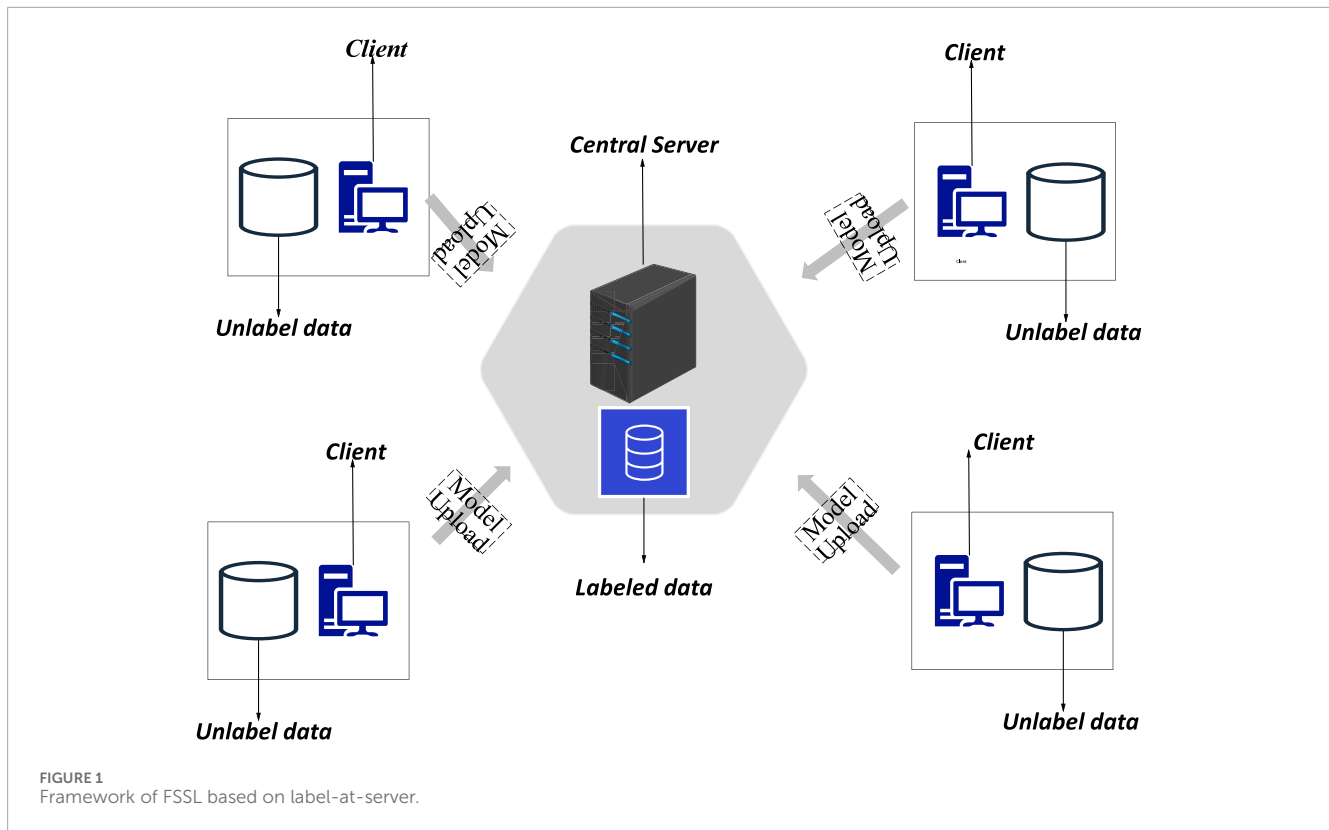
- We apply MMD to align the feature spaces of all clients to alleviate the influence of statistical distribution heterogeneity across clients. Subsequently, the aligned feature spaces are used to enhance the efficiency of aggregating local models.
- We adopt a dual knowledge distillation mechanism to improve the effectiveness of global training. On the server side, the knowledge distillation mechanism is applied to distill knowledge from the aggregated global model to the student model. On the client side, the knowledge distillation mechanism is applied to infuse global knowledge into local training.
- We employ two public datasets to evaluate the performance of FedAlign. The evaluation results demonstrate that FedAlign offers a distinct advantage in terms of efficiency compared with traditional FSSL models.

The remaining sections of this article are organized as follows: Section 2 reviews relevant literature. Section 3 elaborates on the details of FedAlign. Section 4 presents the experimental verification and analysis of FedAlign. Section 5 presents the conclusion drawn from the findings.

2 Related work

A CPSS is an entity that integrates computing, networking, and physical processes to implement integration and dynamic interaction between cyberspace and the physical world [23]. In [24], Javier et al. proposed a complete real-world CPSS implementation cycle including processing and interpretation. In [25], Fei et al. utilized machine learning to address physical layer authentication for a CPSS. In [26], Charles et al. proposed an integrated toolchain to achieve architectural modeling of a CPSS with learning-enabled components. Although there is an abundance of research outcomes in CPSSs, they still face challenges such as data silos and a lack of labeled data [27].

In recent years, FSSL has received more attention. In [10], FedIRM integrates consistency regularization in SSL with FL. FedIRM applies an inter-client relationship matching scheme based on an improved consistency regularization mechanism to enhance the relationship between labeled clients and unlabeled clients.

FIGURE 1
Framework of FSSL based on label-at-server.

In [28], Bdair proposed an FSSL model, denoted as FedPerl. FedPerl introduces a peer anonymous learning mechanism to FSSL. Compared with the mean and standard deviation of parameters from different layers of neural networks, FedPerl constructs a similarity matrix. Based on the similarity matrix, FedPerl can generate high-quality pseudo-labels. In [19], RSCFed applies a distance-reweighted model to generate the global model. Based on the global model, RSCFed aggregates sub-consensus models to update the global model. In [16], Liu et al. introduce representation alignment to FL. By aligning local features with class proxies of the labeled data on the server side, the model effectively mitigates the bias caused by non-IID. In [29], FedRVR constructs a relation-guided multi-functional regularization framework. Based on the constructed framework, FedRVR utilizes model-guided regularization and data-guided regularization to encourage local models to maintain predictive invariance. In [30], Wang et al. propose an FSSL model for alleviating the influence of unreliable data. During the training process, the model utilizes a trustworthy global teacher model to guide local student models to deeply explore the features of unreliable data.

3 Proposed methods

3.1 Problem definition

In this article, we focus on federated semi-supervised learning under the label-at-server scenario for a CPSS, assuming a label-at-server scenario with one server and N clients $\mathcal{N} = \{1, 2, \ldots, N\}$. On the server side, the labeled data are denoted as $D_l = \{(x_i, y_i)\}_{i=1}^{L}$. For
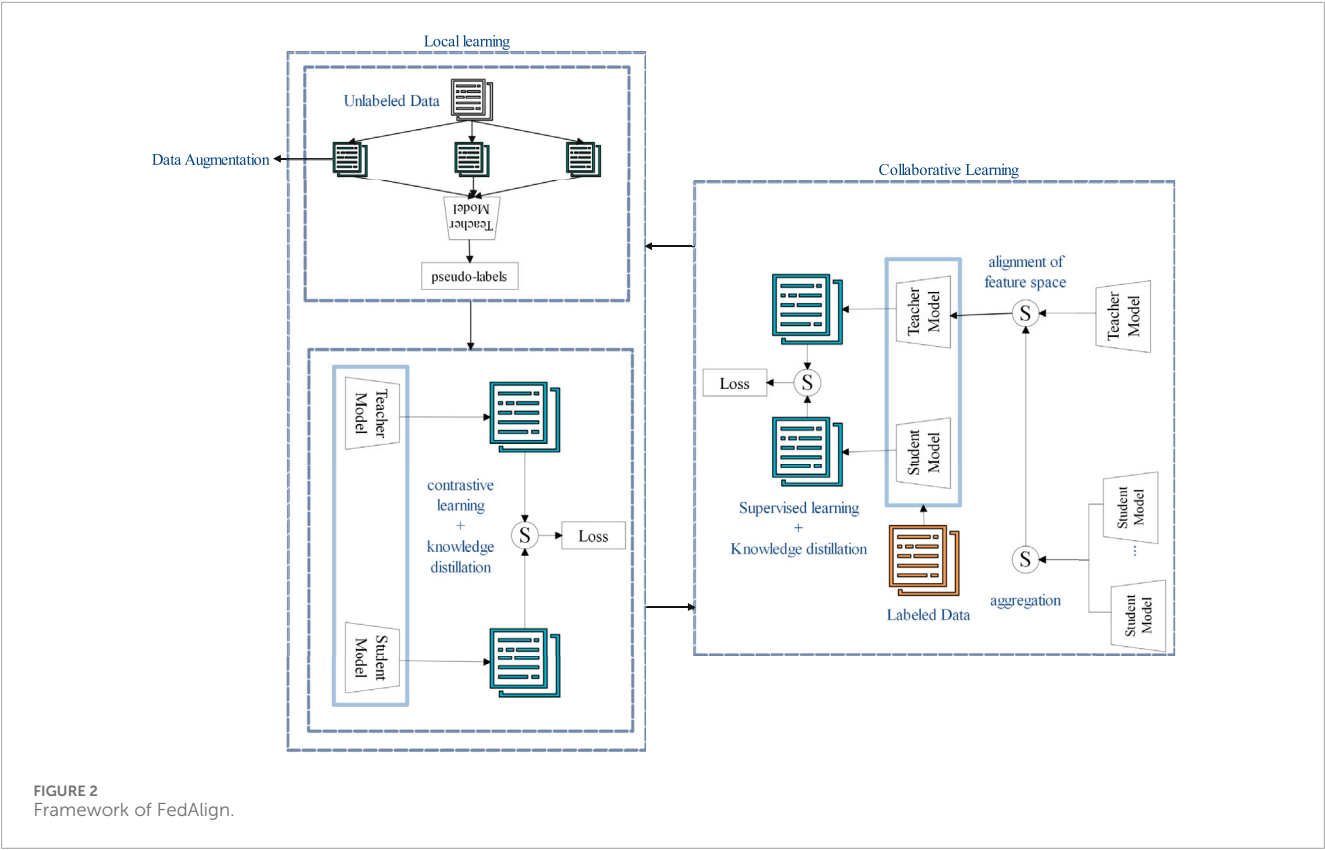
client $j \in \mathcal{N}$, the unlabeled data are denoted as $D_u^j = \{(u_i)\}_{i=1}^{U}$. In the label-at-server scenario, clients apply self-supervised learning to train local models while the server aggregates local models into the global model and utilizes the labeled data to optimize the global model. The final target is to minimize the loss function defined in Equation 1:

$$\mathcal{L}(\omega) = l_{server} + \sum l_{client}, \tag{1}$$

where l_{server} is the loss function on the server side. l_{client} is the loss function of local training on the client side.

3.2 Framework of FedAlign

We propose a novel FSSL model, denoted as FedAlign, under the label-at-server scenario. The framework of FedAlign is shown in Figure 2. FedAlign employs a dual knowledge distillation mechanism, thus adopting a teacher–student framework on both the client and server sides. The teacher model is used as a carrier of global knowledge. The student model is utilized to learn local knowledge. On the client side, the teacher model is updated using the global model sent from the server side. The client utilizes local unlabeled data to train the local student model. Subsequently, the knowledge distillation mechanism is applied to transfer global knowledge from the teacher model to the student model. Finally, the trained student model is uploaded to the server side. On the server side, all the local models received from clients are aggregated to update the global model. The MMD is applied to align feature spaces between the aggregated global model from the current

FIGURE 2
Framework of FedAlign.

communication and that from the previous communication. The aligned feature space is used to optimize the teacher model on the server side. Subsequently, the knowledge distillation mechanism is applied to transfer global knowledge from the teacher model to the student model. The local labeled data are used to optimize the student model. Finally, the optimized student model is sent to clients. The objectives of this mechanism are shown as follows.

- A knowledge distillation mechanism on the client sides can reduce the influence of local data bias on model training to prevent the model from over-adapting to local data.
- A knowledge distillation mechanism on the server side guides the aggregated global model to incorporate local knowledge from client sides while preserving globally generalizable features.

FedAlign is composed of the Local Learning module on the client side and the Collaborative Learning module on the server side. The detailed overview is illustrated in Figure 2.

3.3 Local learning module

The local learning module employs contrastive learning and a knowledge distillation mechanism to learn local knowledge on the client side. Additionally, pseudo-labels for unlabeled data are used to improve the efficiency of local training. The overview of local training is shown in Algorithm 1.

```
1:Input: Client-side dataset Du, batch size B.
2:Output: trained student model
3:for clients parallel training do
4:    receives the global model from server side.
5:    the global model is applied to update the
teacher model.
6:    the client employs Equation 4 to produce
high-quality pseudo-labels for unlabeled data.
7:    the client utilizes unlabeled data with
high-quality pseudo-labels to train the
student model.
8:    the client applies contrastive learning to
train the student model.
9:    the client applies the knowledge
distillation mechanism to transfer global
knowledge from teacher model to student model.
10:    client send trained student model to
server side.
11:end for
```

Algorithm 1. Local training.

3.3.1 Generation of pseudo-labels

In the label-at-server scenario, the data on the client side are exclusively unlabeled data. Thus, during the training process, the client side lacks supervised signals. To compensate for the lack of supervised signals, FedAlign generates pseudo-labels for

unlabeled data on the client side. FSSL is based on the assumption that samples with similar features exhibit consistent predictions. Therefore, FedAlign adopts a pseudo-label generation mechanism based on consistency verification to generate high-quality pseudo-labels for unlabeled data.

The details of pseudo-label generation are shown as follows. First, FedAlign performs data augmentation on unlabeled data to generate multiple augmented samples. Subsequently, FedAlign utilizes the teacher model to predict these augmented samples. The client uses these prediction results to assess the consistency of unlabeled data. Finally, this consistency is utilized to generate the pseudo-labels with high quality. The definition of obtaining consistency is defined from Equations 2-4. On the client side, the teacher model is constructed based on the global model received from the server side. Thus, the generation of pseudo-labels is based on global knowledge and is more robust to data variations.

$$\hat{y}(x) = f(x; \theta_{tea}), \tag{2}$$

$$\hat{y}_v^i(x) = \underset{c \in \{1,2,\ldots,C\}}{argmax} f(aug_i(x); \theta_{tea}), \tag{3}$$

$$consist(x) = \sum_{i=1}^{K} 1\left\{\hat{y}_v^i(x) = \hat{y}(x)\right\} > r, \tag{4}$$

where $f(x; \theta_{tea})$ denotes the prediction of the teacher model for the unlabeled data x and θ_{tea} denotes the parameter of the teacher model on the client side. $\{1, 2, \ldots, C\}$ is the set of category. $aug_i(\cdot)$ is the ith augmentation. $1(\cdot)$ is the indicator function. r denotes a hyperparameter. If the consistency of the unlabeled data x ($consist(x)$) is greater than r, the pseudo-label of the unlabeled data x is of high quality. Finally, the client applies pseudo-labels to optimize the student model. The loss function is defined as Equation 5:

$$\mathcal{L}_{sup} = H(f(x; \theta_{stu}), \hat{y}) = -\sum_{j=1}^{m} \hat{y}^j \cdot \log(f(x; \theta_{stu})^i), \tag{5}$$

where x is unlabeled data with high-quality pseudo-label. $\hat{y}$ is the one-hot encoding of the pseudo-label for x. $\hat{y}^i$ is the ith element in $\hat{y}$.

3.3.2 Local feature learning

In the local feature learning module, FedAlign integrates contrastive learning with a knowledge distillation mechanism to mine self-supervised information from unlabeled data. Based on the mined information, FedAlign can effectively extract discriminative representations of general features. So, FedAlign not only alleviates the issue of insufficient supervised signals on the client side but also enhances the robustness of clients to non-IID data.

For FedAlign, contrastive learning is applied to learn stable and discriminative features from unlabeled data. The target is to reduce feature shifts caused by non-IID data. Contrastive learning is a branch of self-supervised learning. It can guide the model to learn inherent structures and discriminative features from unlabeled data based on contrastive relationships between similar and dissimilar samples. For FedAlign, clients apply contrastive learning to extract inherent features from unlabeled data and maintain the robustness

for non-essential variations. The loss function of contrastive learning applied in FedAlign is defined as Equation 6:

$$\mathcal{L}_{contrast} = -\log \frac{\exp\left(sim\left(f(aug_a(x); \theta_{stu}), f(aug_b(x); \theta_{stu})\right)/\tau\right)}{\sum_{\acute{x} \in \mathcal{B}, \acute{x} \neq x} \exp\left(sim\left(f(aug_a(x); \theta_{stu}), f(\acute{x}; \theta_{stu})\right)/\tau\right)}, \tag{6}$$

where θ_{stu} is the parameter of student model in client. x_a and x_b are two augmentations for unlabeled data x. $sim(\cdot)$ is similarity function.

For FedAlign, there are three targets of knowledge distillation. First, the knowledge distillation mechanisms utilized to transfer global knowledge from the teacher model to the student model on the client side. Second, for knowledge distillation, clients apply the output of the teacher model as a soft supervised signal to prevent overfitting to local data on the client side. Third, the knowledge distillation mechanism can align the output of the student model with that of the teacher model to mitigate notable differences between the data distribution on the client side and the global data distribution. Thus, these targets of knowledge distillation compensate for the limitations of non-IID data on the client side and prevent overfitting of student models to local data. The loss function of knowledge distillation is shown as Equation 7:

$$\mathcal{L}_{distill} = \frac{1}{2} \cdot \left\{\mathcal{L}_{KL}\left(f(aug_a(x); \theta_{stu}), f(aug_a(x); \theta_{tea})\right)\right.$$
$$\left. + \mathcal{L}_{KL}\left(f(aug_b(x); \theta_{stu}), f(aug_b(x); \theta_{tea})\right)\right\}, \tag{7}$$

where $\mathcal{L}_{KL}$ is Kullback–Leibler divergence. θ_{stu} is a parameter of the student model. θ_{tea} is a parameter of the teacher model. $aug_a(x)$ and $aug_b(x)$ are two augmentations for unlabeled data x. The total loss function in local learning module is defined as Equation 8.

$$\mathcal{L}_{client} = \lambda_1 \cdot \mathcal{L}_{sup} + \lambda_2 \cdot \mathcal{L}_{constrast} + \lambda_3 \cdot \mathcal{L}_{distill}. \tag{8}$$

3.4 Collaborative learning module

The collaborative learning module is able to learn global information by fusing local model aggregation, space alignment, and the knowledge distillation mechanism. In the tth communication round, for weight aggregation, FedAlign introduces the prediction accuracy of local models received from the client sides on the labeled data as the weight for aggregation to update the teacher model on the server side. For space alignment, FedAlign applies MMD to measure the difference between the student models trained in the $(t-1)$th communication round and the updated teacher model on the server side to fine-tune the teacher model. Subsequently, the knowledge distillation mechanism and supervised learning are applied to train the student model on the server side. Finally, the student model is sent to the client sides. The overview of the collaborative learning module in the tth communication round is shown in Algorithm 2.

3.4.1 Aggregation of local models

The aggregation methods in traditional FL are based on the amount of data on clients while ignoring the distribution discrepancy between the learned information and the global information. Thus, FedAlign utilizes the prediction accuracy of the local models received from the client sides on the labeled data to evaluate the learning quality of the local models.

```
1:  Input: Server-side dataset D₁, batch size B,
the student model trained in the t−1th
communication round.
2:  Output: the parameter of the trained
student model.
3:  receives the local model from the
client sides.
4:  assesses the prediction accuracy of the local
models using labeled data.
5:  employs Equation 12 to aggregate local models
to generate the global model. Then, the generated
model is applied to update the teacher model.
6:  applies MMD for aligning feature space between
the updated teacher model and student model
trained in the t−1th communication round to
fine-tune the teacher model.
7:  employs the knowledge distillation mechanism
to convey global knowledge from the teacher model
to the student model.
8:  applies supervised learning based on cross
entropy to optimize the student model.
9:  store and send the trained student model to
selected clients.
```

Algorithm 2. Collaborative learning in the t-th communication round.

- **Local model with high accuracy:** The local models received from the client sides with high accuracy indicate that the knowledge learned by the clients is more consistent with the global data distribution. Although non-IID data exist in the data across clients, these local models still produce reliable predictions. Thus, it is more valuable to utilize such high-accuracy local models for updating the global model.
- **Local model with low accuracy:** Local models with low accuracy indicate that data on the client have an extremely skewed local distribution or suffers from overfitting. This is due to a significant deviation between the learned information and the global data distribution. If excessive weight is assigned to such local models, it will cause the global model to deviate from the global data distribution.

In conclusion, the aggregation of local models is defined from Equations 9-12. Subsequently, the parameters of the aggregated model are utilized to update the teacher model on the server side.

$$\acute{a}_c = \frac{a_c - \min(a)}{\max(a) - \min(a)}, \qquad (9)$$

$$\acute{s}_c = \frac{s_c - \min(s)}{\max(s) - \min(s)}, \qquad (10)$$

$$\omega_c = \frac{\lambda_1 \cdot \acute{a}_c + \lambda_2 \cdot \acute{s}_c}{\sum_{i \in \mathcal{S}} \lambda_1 \cdot \acute{a}_i + \lambda_2 \cdot \acute{s}_i}, \qquad (11)$$

$$\theta_{agg} = \sum_{c \in \mathcal{S}} \omega_c \cdot \theta_c, \qquad (12)$$

where a_c is the prediction accuracy of client c on the labeled data. s_c is the amount of data in client c. $min(a)$ and $max(a)$ are the minimum value and maximum value of the prediction accuracy across all clients in one communication round. $min(s)$ and $max(s)$ are the minimum value and maximum value of the amount of data across all clients in one communication round. $\mathcal{S}$ is a set of clients selected for communication. θ_c is the parameter of local model received from client c.

3.4.2 Space alignment

FedAlign applies MMD to align the feature spaces between the student model obtained in the previous communication round and the teacher model on the server side. Although FedAlign updates the teacher model by aggregating the received local models on the server side, as shown in Section 3.4.1, the teacher model cannot guarantee the consistency of the semantic understanding of data. MMD can measure the feature distribution discrepancy between the updated teacher model and the student model trained in the previous communication round. The student model carries the global knowledge learned in the previous communication round. Based on the discrepancy, the server aligns the feature distribution of the updated teacher model with that of the global model learned in the previous communication round. The advantages of this alignment mechanism are as follows.

- Alleviating the distribution shift caused by non-IID. In a federated learning environment, non-IID data across clients lead to fluctuations in the feature distribution of the global model across rounds. MMD directly measures the distribution discrepancy between the feature spaces of the updated teacher model and the student model trained in the previous communication round. By minimizing this discrepancy, the teacher model inherits the feature distribution characteristics of the previous global model in order to avoid knowledge fragmentation caused by data heterogeneity and enhance the robustness of the model to heterogeneously distributed data.
- Enhancing the aggregation of global knowledge. The traditional generation of local models has already integrated common knowledge from multiple clients. For FedAlign, MMD alignment essentially enables the new teacher model to inherit this common knowledge. Compared with training solely on the local data on the server side, MMD allows the teacher model to absorb more comprehensive information across clients in order to mitigate local data bias and enhance the completeness and effectiveness of knowledge transfer.

In FSSL, based on the label-at-server scenario, features extracted by local models are high-dimensional and non-linear. Thus, it is difficult to describe these high-dimensional and non-linear distributions using simple probabilistic models. FedAlign employs MMD to map the features extracted by the two models to the reproducing kernel Hilbert space (RKHS) via multi-bandwidth Gaussian kernels. Using these mapped features, MMD utilizes the distance between the means of the feature distributions extracted by the updated teacher model and the global model from the previous communication round to measure the discrepancy between these two feature distributions. Based on the discrepancy, FedAlign can effectively quantify and align the feature distributions of these

models in order to alleviate distribution shift in non-IID scenarios of FSSL while adapting to the constraints of limited labeled data. The loss function of MMD in FedAlign is defined as Equations 13, 14:

$$k(f_1, f_2) = \sum_{i=1}^{M} \exp\left(-\frac{\|f_1 - f_2\|^2}{2\sigma_i^2}\right), \quad (13)$$

$$\mathcal{L}_{MMD} = \frac{1}{n^2} \cdot \sum_{i=1,j} k\left(f_T^i, f_T^j\right) + \frac{1}{n^2} \cdot \sum_{i,j=1} k\left(f_S^i, f_S^j\right) - \frac{2}{n^2} \sum_{i=1,j=1} k\left(f_T^i, f_S^j\right), \quad (14)$$

where $\{f_T^1, f_T^2, \ldots, f_T^n\}$ is a feature extracted by the updated teacher model based on labeled data. $\{f_S^1, f_S^2, \ldots, f_S^n\}$ is a feature extracted by the student model trained in the previous communication round based on labeled data. M is the number of kernels.

3.4.3 Global knowledge transmission

FedAlign employs the knowledge distillation mechanism to effectively transmit global knowledge from the teacher model to the student model. For FedAlign, after completing the feature space alignment on the server side, knowledge distillation is applied to enable the student model to further learn the teacher's refined knowledge about complex samples based on a consistent feature distribution with the teacher model. Additionally, the knowledge distillation mechanism can correct any hidden residual biases in the student model to enhance the generalization ability and robustness of the global model.

For FedAlign, the server applies the predicted results from the fine-tuned teacher model on the labeled data to classify the labeled data into two distinct categories, namely, high-quality data and low-quality data. High-quality data are data for which the predicted results are correct. Low-quality data are those for which the predicted results are incorrect. The loss function of the knowledge distillation mechanism in FedAlign is defined as Equation 15:

$$\mathcal{L}_{distill,server} = \lambda_{s1} \cdot \mathcal{L}_{KL}\left(f(x; \theta_{tea}), f(x; \theta_{stu})\right) + \lambda_{s2} \cdot \mathcal{L}_{KL}\left(f(\acute{x}; \theta_{tea}), f(\acute{x}; \theta_{stu})\right), \quad (15)$$

where x is data belonging to high-quality data. $\acute{x}$ is data belonging to low-quality data. $\mathcal{L}_{KL}$ is Kullback–Leibler divergence. On the server side, FedAlign also applies supervised learning based on cross entropy to optimize the student model. The loss function of supervised learning is defined as Equation 16:

$$\mathcal{L}_{sup-server} = H\left(f(x; \theta_{stu}), y\right) = -\sum_{i=1}^{m} y^i \cdot \log\left(f(x; \theta_{stu})^i\right), \quad (16)$$

where x is labeled data. y is one-hot encode of label of x. θ_{stu} is the parameter of the student model. The total loss function for the student model is defined in Equation 17:

$$\mathcal{L}_{stu-server} = \lambda_{dis} \cdot \mathcal{L}_{distill,server} + \lambda_{sup} \cdot \mathcal{L}_{sup-server}, \quad (17)$$

where λ_{dis} and λ_{sup} are hyperparameters.

4 Experimental evaluation

4.1 Experimental setup

In this section, we utilize two public datasets, CIFAR-10 and Fashion-MNIST, to evaluate the efficiency of FedAlign.

- **CIFAR-10** [31]**:** CIFAR-10 is a well-known public dataset dedicated to image classification tasks. This dataset is utilized as a benchmark to evaluate the performance of models in terms of feature extraction and generalization for small-scale color images. CIFAR-10 comprises 60,000 RGB color images, divided into 10 categories, with a resolution of 32×32 pixels, where the pixel values span from 0 to 255. Regarding its data split, the training dataset in CIFAR-10 contains 50,000 images, while the test dataset has 10,000 images.
- **Fashion-MNIST** [32]**:** Fashion-MNIST is a well-established public image dataset, and it is utilized to replace the MNIST dataset for alleviating the issue of inflated model performance. This dataset contains 70,000 grayscale images belonging to 10 categories, with a resolution of 28×28 pixels. The training dataset in Fashion-MNIST includes 60,000 images, while the test dataset in Fashion-MNIST includes 10,000 images.

In order to effectively train FedAlign, we adopt stochastic gradient descent (SGD) with an initial learning rate of 0.0001 for model optimization. The total number of communication rounds is 300, and the number of clients is 100. In each communication round, the server selects 10 clients to participate in global model training. Regarding training epochs, the warm-up phase on the server side is set to 10 epochs, while each client side performs 10 training epochs. MobileNetV2 is adopted as the backbone model. The amount of labeled data N is set to 500 and 1,000. We apply the Dirichlet distribution ($\alpha = 1.0$) to simulate the non-IID distribution across clients.

4.2 Baseline method

We select four models as baselines. The details of those baselines are as follows.

- **SL-Server:** This model only uses labeled data on the server side to train the global model.
- **SEL-FedAvg:** On the client side, this model uses contrastive learning to train the local model. On the server side, it utilizes FedAvg to aggregate local model updates. Subsequently, the server uses labeled data to optimize the aggregated global model.
- **FedMatch** [33]**:** FedMatch uses consistency regularization to train the local model on the client side. On the server side, it leverages improvements to the aggregation mechanism of FedAvg to generate the global model.
- **FedCon** [34]**:** FedCon applies a contrastive network and a novel two-output model to train the local model on the client side.

4.3 Overall result

To evaluate the training efficiency of FedAlign, we trained it under non-IID and IID scenarios. The training results are shown in Figure 3. As illustrated in Figure 3, the test accuracy continually improves as the number of communication rounds increases. When the number of communication rounds exceeds 150,
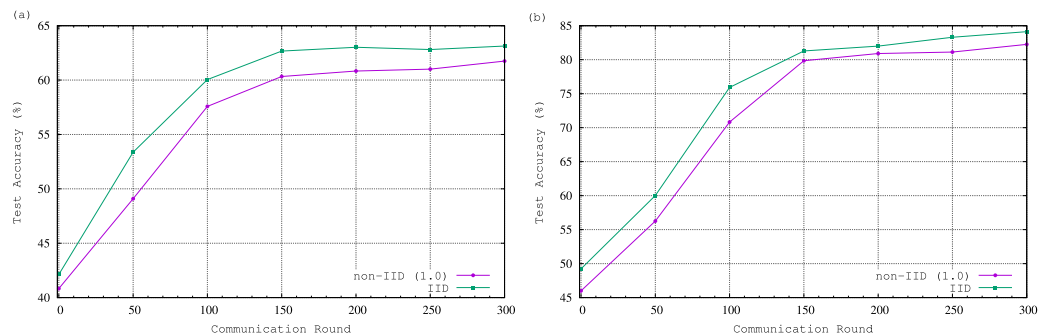
FIGURE 3
Variations of test accuracy with increasing numbers of communication rounds ($N = 500$). **(a)** Test accuracy of CIFAR-10. **(b)** Test accuracy of fashion-MNIST.
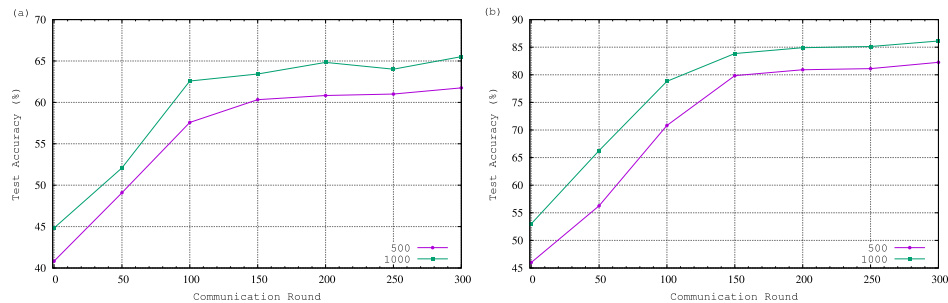

FIGURE 4
Variations of test accuracy with different amounts of labeled data ($\alpha = 1.0$). **(a)** Compare of different amount of labeled sample for CIFAR-10. **(b)** Compare of different of labeled sample for Fashion-MNIST.

TABLE 1 Highest accuracy results compared for CIFAR-10.

Model	Highest accuracy (%)			
	500		1,000	
	IID	Non-IID	IID	Non-IID
SL-Server	50.15	49.36	56.82	54.43
SEL-FedAvg	52.43	50.62	57.52	55.95
FedMatch	53.62	52.76	68.22	67.14
FedCon	58.43	54.84	60.24	63.14
FedAlign (ours)	63.13	61.75	67.43	65.52

TABLE 2 Highest accuracy results compared for Fashion-MNIST.

Model	Highest accuracy (%)			
	500		1,000	
	IID	Non-IID	IID	Non-IID
SL-Server	61.24	40.41	66.34	52.13
SEL-FedAvg	68.28	62.94	68.25	65.36
FedMatch	81.92	81.59	85.49	83.24
FedCon	79.43	80.24	83.24	81.92
FedAlign (ours)	84.12	82.25	87.61	86.12

the improvement speed begins to plateau. When the number of communication rounds reaches 300, the test accuracy under the non-IID and IID scenarios is 63.13% and 61.75%, respectively.

In order to evaluate the influence of the amount of labeled data on the server side on training efficiency, we trained FedAlign under non-IID scenarios with different volumes of server-side labeled data. The experimental results are shown in Figure 4. As illustrated in Figure 4, although the test accuracy of FedAlign

continually improves as communication rounds increase, the model trained with 1,000 labeled samples achieves higher test accuracy than that trained with 500 labeled samples. This indicates that increasing the amount of labeled data on the server side can mitigate the adverse impact of non-IID data distribution on training performance.

The results of FedAlign compared with all the baselines are shown in Tables 1, 2. These results demonstrate that FedAlign
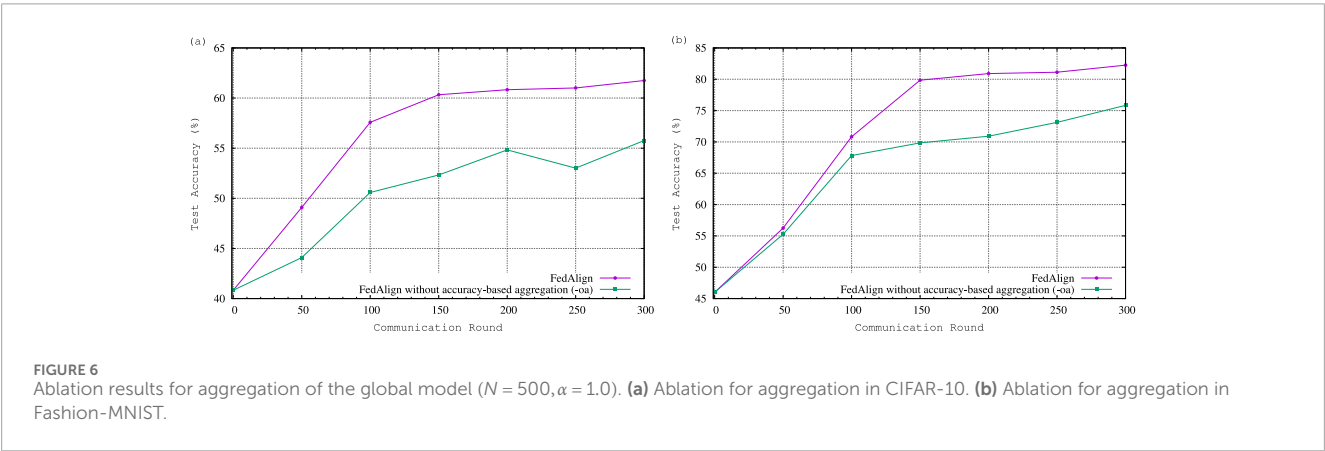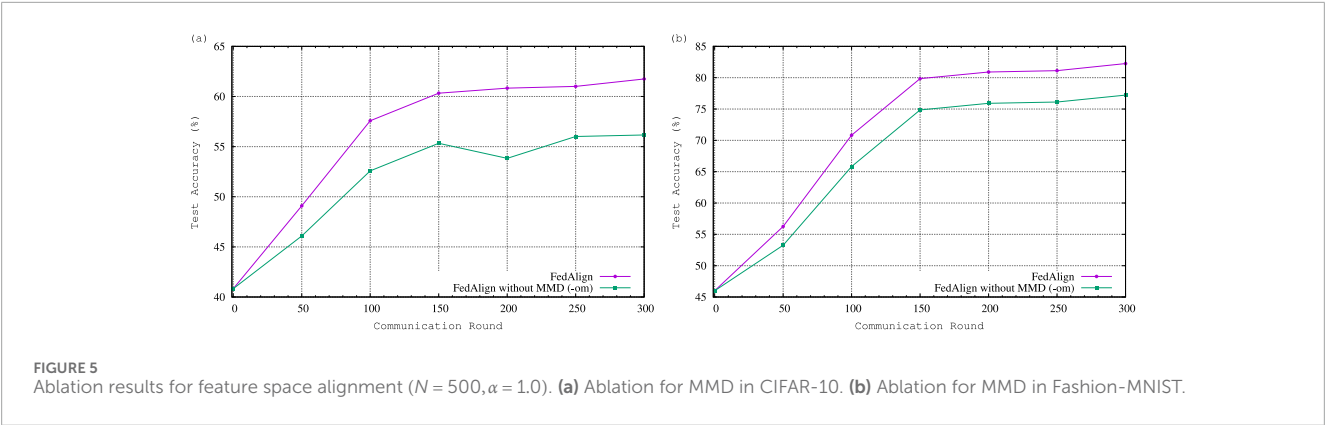
FIGURE 5
Ablation results for feature space alignment ($N = 500, \alpha = 1.0$). **(a)** Ablation for MMD in CIFAR-10. **(b)** Ablation for MMD in Fashion-MNIST.

FIGURE 6
Ablation results for aggregation of the global model ($N = 500, \alpha = 1.0$). **(a)** Ablation for aggregation in CIFAR-10. **(b)** Ablation for aggregation in Fashion-MNIST.

TABLE 3 Ablation results.

Labeled data	Setting	Highest accuracy (%)					
		CIFAR-10			Fashion-MNIST		
		FedAlign	-om	-oa	FedAlign	-om	-oa
500	IID	63.13	58.72	56.58	84.12	79.63	76.92
	Non-IID	61.75	56.16	55.76	82.25	77.22	75.84
1,000	IID	67.43	62.37	59.37	87.61	84.32	81.19
	Non-IID	65.52	60.92	57.17	86.12	80.38	79.22

achieves higher test accuracy than all baselines. SL-Server only utilizes labeled data on the server side to train the global model. Thus, SL-Server cannot leverage local knowledge from the client side, and consequently, its test accuracy is lower than that of all other models. SEL-FedAvg adopts FedAvg's aggregation mechanism to construct the global model. Although SEL-FedAvg employs semi-supervised learning to leverage local knowledge from the client side, its aggregation of local model updates received from clients overlooks the impact of non-IID data distribution. Thus, SEL-FedAvg tends to converge to a local optimum. FedMatch applies consistency regularization to learn local knowledge from the client side and leverages pseudo-labels to optimize the local model. However, in the early stages of training, the low-quality global model produces a large number of low-quality pseudo-labels, which hinder the model's training efficiency. Although FedMatch leverages the training frequency of clients to aggregate the global model, the non-IID issue is not effectively mitigated. FedCon integrates a contrastive learning network into local training but overlooks feature space alignment of data across clients. Thus, FedCon cannot mitigate the adverse impact of non-IID on global model training.

For FedAlign, MMD is employed on the server side to align the feature spaces of data across clients. Additionally, the server utilizes the prediction accuracy of received local models to assess the data distribution of each client. Based on these evaluation results, the server constructs the global model. Thus, FedAlign can effectively leverage local knowledge from the client side. As a result, FedAlign outperforms all baselines in terms of training efficiency and test accuracy.

4.4 Analysis and ablation study

In this section, we analyze the influence of local model aggregation and feature space alignment on FedAlign. Thus, we compare FedAlign with its variant without MMD (-om) to analyze the impact of feature space alignment on FedAlign. Additionally, we compare FedAlign with its variant without accuracy-based aggregation (-oa) to evaluate the effect of prediction accuracy-driven aggregation.

Regarding the influence of MMD, the comparison results are shown in Figure 5. As illustrated in Figure 5, the test accuracy of FedAlign is higher than that of its variant without MMD. This indicates that lacking feature space alignment causes the global model training to get trapped in a local optimum. This is because FedAlign can effectively quantify and align the feature distributions of all clients to alleviate distribution shift in non-IID scenarios of FSSL while adapting to the constraint of limited labeled data.

Regarding the influence of prediction accuracy-driven aggregation, the comparison results are shown in Figure 6. As illustrated in Figure 6, the test accuracy of FedAlign is higher than that of its variant without accuracy-based aggregation (-oa). This indicates that aggregation without prediction accuracy weighting results in a low-quality global model. Subsequently, the low-quality global model generates low-quality pseudo-labels on the client side. The ablation study results are summarized in Table 3.

5 Conclusion

In this article, we present a novel FSSL model based on the label-at-server framework for a distributed CPSS, denoted as FedAlign. For FedAlign, MMD is employed on the server side to align the feature spaces of client-side data; meanwhile, the prediction accuracy of local models on the server's labeled data is leveraged to weight the aggregation of the global model, and a knowledge distillation mechanism is utilized to distill and transfer global knowledge from the teacher model to the student model. On the client side, FedAlign integrates knowledge distillation with contrastive learning to train local models. Thus, FedAlign can effectively train the global model without sharing raw data. As a result, FedAlign can effectively address the CPSS dilemmas of data silos and label scarcity. Ultimately, we evaluate FedAlign's performance using two public datasets, and the experimental results demonstrate that FedAlign outperforms other baseline models in terms of both performance and efficiency.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

Author contributions

ZD: Writing – review and editing, Formal analysis, Writing – original draft, Data curation, Investigation, Methodology. HY: Writing – review and editing, Investigation. WX: Writing – review and editing. YX: Writing – review and editing. QW: Writing – review and editing. QC: Writing – review and editing. ZQ: Investigation, Writing – review and editing, Methodology. DC: Writing – review and editing, Investigation.

Funding

Conflict of interest

Author QC was employed by Accelink Technologies Co., Ltd.

The remaining author(s) declared that this work was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declared that generative AI was not used in the creation of this manuscript.

Any alternative text (alt text) provided alongside figures in this article has been generated by Frontiers with the support of artificial intelligence and reasonable efforts have been made to ensure accuracy, including review by the authors wherever possible. If you identify any issues, please contact us.

Publisher's note

References

1. Zhou Y, Yu FR, Chen J, Kuo Y. Cyber-physical-social systems: a state-of-the-art survey, challenges and opportunities. *IEEE Commun Surv & Tutorials* (2019) 22:389–425. doi:10.1109/comst.2019.2959013

2. Zeng J, Yang LT, Lin M, Ning H, Ma J. A survey: cyber-physical-social systems and their system-level design methodology. *Future Generation Comput Syst* (2020) 105:1028–1042. doi:10.1016/j.future.2016.06.034

3. Xiao P, Chen D, Qin Z, Cao M, Chen R. Edge-adaptive dynamic scalable convolution for efficient remote mobile pathology analysis. In: *ACM transactions on autonomous and adaptive systems* (2025).

4. Luo X, Wu Q, Wang Y, Zhang K, Dai HN, Chen D. Tmae: entropy-aware masked autoencoder for low-cost traffic flow map inference. *IEEE Internet Things J* (2025) 12:27255–27268. doi:10.1109/jiot.2025.3563583

5. Ale L, Zhang N, King SA, Chen D. Empowering generative ai through mobile edge computing. *Nat Rev Electr Eng* (2024) 1:478–486. doi:10.1038/s44287-024-00053-6

6. Qian F, Tang Y, Yu X. The future of process industry: a cyber–physical–social system perspective. *IEEE Trans Cybernetics* (2023) 54:3878–3889. doi:10.1109/TCYB.2023.3298838

7. Zhou X, Liang W, Ma J, Yan Z, Wang KIK. 2d federated learning for personalized human activity recognition in cyber-physical-social systems. *IEEE Trans Netw Sci Eng* (2022) 9:3934–3944. doi:10.1109/tnse.2022.3144699

8. Shi P, Winter JS, Zhang B. Governance of privacy protection: how laws will be adopted to address new technologies? (2021).

9. Van Engelen JE, Hoos HH. A survey on semi-supervised learning. *Machine Learning* (2020) 109:373–440. doi:10.1007/s10994-019-05855-6

10. Liu Q, Yang H, Dou Q, Heng PA. Federated semi-supervised medical image classification via inter-client relation matching. In: *International conference on medical image computing and computer-assisted intervention*. Springer (2021). p. 325–335.

11. Long Z, Che L, Wang Y, Ye M, Luo J, Wu J, et al. Fedsiam: towards adaptive federated semi-supervised learning. arXiv preprint arXiv:2012.03292 (2020).

12. Pasandideh S, Pereira P, Gomes L. Cyber-physical-social systems: taxonomy, challenges, and opportunities. *IEEE Access* (2022) 10:42404–42419. doi:10.1109/access.2022.3167441

13. Zhang Z, Zhang F, Xiong Z, Zhang K, Chen D. Lsia3cs: deep-reinforcement-learning-based cloud–edge collaborative task scheduling in large-scale iiot. *IEEE Internet Things J* (2024) 11:23917–23930. doi:10.1109/jiot.2024.3386888

14. Chen D, Liao Z, Xie Z, Chen R, Qin Z, Cao M, et al. Mfsse: multi-keyword fuzzy ranked symmetric searchable encryption with pattern hidden in mobile cloud computing. *IEEE Trans Cloud Comput* (2024) 12:1042–1057. doi:10.1109/tcc.2024.3430237

15. Sun J, Chen D, Zhang N, Xu G, Tang M, Nie X, et al. A privacy-aware and traceable fine-grained data delivery system in cloud-assisted healthcare iiot. *IEEE Internet Things J* (2021) 8:10034–10046. doi:10.1109/jiot.2020.3048976

16. Liu H, Mi Y, Tang Y, Guan J, Zhou S. Boosting semi-supervised federated learning by effectively exploiting server-side knowledge and client-side unconfident samples. *Neural Networks* (2025) 188:107440. doi:10.1016/j.neunet.2025.107440

17. Naeem F, Ali M, Kaddoum G. Federated-learning-empowered semi-supervised active learning framework for intrusion detection in zsm. *IEEE Commun Mag* (2023) 61:88–94. doi:10.1109/mcom.001.2200533

18. Jin Y, Liu Y, Chen K, Yang Q. Federated learning without full labels: a survey. arXiv preprint arXiv:2303.14453 (2023). doi:10.48550/arXiv.2303.14453

19. Liang X, Lin Y, Fu H, Zhu L, Li X. Rscfed: random sampling consensus federated semi-supervised learning. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2022). p. 10154–10163.

20. Chen D, Liao Z, Chen R, Wang H, Yu C, Zhang K, et al. Privacy-preserving anomaly detection of encrypted smart contract for blockchain-based data trading. *IEEE Trans Dependable Secure Comput* (2024) 21:4510–4525. doi:10.1109/tdsc.2024.3353827

21. Li M, Li Q, Wang Y. Class balanced adaptive pseudo labeling for federated semi-supervised learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition; 17-24 June 2023; Vancouver, BC, Canada. IEEE (2023). p. 16292–16301.

22. Qiu L, Cheng J, Gao H, Xiong W, Ren H. Federated semi-supervised learning for medical image segmentation via pseudo-label denoising. *IEEE Journal Biomedical Health Informatics* (2023) 27:4672–4683. doi:10.1109/JBHI.2023.3274498

23. Wang FY. The emergence of intelligent enterprises: from cps to cpss. *IEEE Intell Syst* (2010) 25:85–88. doi:10.1109/mis.2010.104

24. Diaz-Rozo J, Bielza C, Larrañaga P. Machine learning-based cps for clustering high throughput machining cycle conditions. *Proced Manufacturing* (2017) 10:997–1008. doi:10.1016/j.promfg.2017.07.091

25. Pan F, Pang Z, Wen H, Luvisotto M, Xiao M, Liao RF, et al. Threshold-free physical layer authentication based on machine learning for industrial wireless cps. *IEEE Trans Ind Inform* (2019) 15:6481–6491. doi:10.1109/tii.2019.2925418

26. Hartsell C, Mahadevan N, Ramakrishna S, Dubey A, Bapty T, Johnson T, et al. Model-based design for cps with learning-enabled components. In: *Proceedings of the workshop on design automation for CPS and IoT* (2019). p. 1–9.

27. Quan MK, Pathirana PN, Wijayasundara M, Setunge S, Nguyen DC, Brinton CG, et al. Federated learning for cyber physical systems: a comprehensive survey. *IEEE Commun Surv & Tutorials* (2025) 1. doi:10.1109/comst.2025.3570288

28. Bdair T, Navab N, Albarqouni S. Fedperl: semi-supervised peer learning for skin lesion classification. In: *International conference on medical image computing and computer-assisted intervention*. Springer (2021). p. 336–346.

29. Yang Q, Chen Z, Peng Z, Yuan Y. Relation-guided versatile regularization for federated semi-supervised learning. *Int J Comput Vis* (2025) 133:1–15. doi:10.1007/s11263-024-02330-1

30. Wang G, Pu C, Fu D, Zhang Y, Yu J, Hou Y. Semi-supervised federated learning fault diagnosis method driven by teacher-student model consistency. *Signal Image Video Process.* (2025) 19:385. doi:10.1007/s11760-025-03935-w

31. Thakkar V, Tewary S, Chakraborty C. Batch normalization in convolutional neural networks—a comparative study with cifar-10 data. In: 2018 fifth international conference on emerging applications of information technology (EAIT); 12-13 January 2018; Kolkata, India. IEEE (2018). p. 1–5.

32. Xiao H, Rasul K, Vollgraf R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747 (2017).

33. Zhang Z, Ma S, Yang Z, Xiong Z, Kang J, Wu Y, et al. Robust semisupervised federated learning for images automatic recognition in internet of drones. *IEEE Internet Things J* (2022) 10:5733–5746. doi:10.1109/jiot.2022.3151945

34. Long Z, Wang J, Wang Y, Xiao H, Ma F. Fedcon: a contrastive framework for federated semi-supervised learning. arXiv preprint arXiv:2109.04533 (2021).