The Thirty-Ninth AAAI Conference on Artificial Intelligence (AAAI-25)

SemiDFL: A Semi-Supervised Paradigm for Decentralized Federated Learning

Xinyang Liu[1,2*], **Pengchao Han**[3*], **Xuan Li**[4], **Bo Liu**[1†]
[1]Shenzhen Institute of Artificial Intelligence and Robotics for Society, China
[2]Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, China
[3]School of Information Engineering, Guangdong University of Technology, China
[4]School of Information Science and Engineering, Southeast University, China
codex.lxy@gmail.com, hanpengchao@gdut.edu.cn, xuanli2003@seu.edu.cn, liubo@cuhk.edu.cn

Abstract

Decentralized federated learning (DFL) realizes cooperative model training among connected clients without relying on a central server, thereby mitigating communication bottlenecks and eliminating the single-point failure issue present in centralized federated learning (CFL). Most existing work on DFL focuses on supervised learning, assuming each client possesses sufficient labeled data for local training. However, in real-world applications, much of the data is unlabeled. We address this by considering a challenging yet practical semi-supervised learning (SSL) scenario in DFL, where clients may have varying data sources: some with few labeled samples, some with purely unlabeled data, and others with both. In this work, we propose SemiDFL, the first semi-supervised DFL method that enhances DFL performance in SSL scenarios by establishing a consensus in both data and model spaces. Specifically, we utilize neighborhood information to improve the quality of pseudo-labeling, which is crucial for effectively leveraging unlabeled data. We then design a consensus-based diffusion model to generate synthesized data, which is used in combination with pseudo-labeled data to create mixed datasets. Additionally, we develop an adaptive aggregation method that leverages the model accuracy of synthesized data to further enhance SemiDFL performance. Through extensive experimentation, we demonstrate the remarkable performance superiority of the proposed DFL-Semi method over existing CFL and DFL schemes in both IID and non-IID SSL scenarios.

Code — https://github.com/ez4lionky/SemiDFL

Introduction

Federated Learning (FL) (McMahan et al. 2017; Kairouz et al. 2021) enables collaborative learning among distributed clients while keeping their data local and preventing "data islands". However, traditional Centralized Federated Learning (CFL) relies on a central server to aggregate model parameters from all clients, which suffers from high communication bottleneck and single-point-failure problems (Liu, Ding, and Lv 2020; Liu and Ding 2021b). Decentralized Federated Learning (DFL) (Beltrán et al. 2023; Sun et al. 2024) addresses this issue by enabling direct communication between clients and their connected neighbors for model aggregation. By removing the central server, DFL alleviates the communication bottleneck of traditional FL and enhances model scalability and robustness (Liu and Ding 2021a).

Existing CFL and DFL approaches are mostly designed for supervised learning, which relies heavily on large quantities of labeled data for model training. However, collecting and labeling extensive datasets in real-world applications is challenging due to time constraints, high costs, and the need for expert knowledge. Furthermore, clients often possess diverse data sources (e.g., labeled, unlabeled, or both) with heterogeneous distributions (Yang et al. 2022). This raises this paper's key question: How can DFL be effective when clients have labeled and unlabeled data sources in highly non-iid scenarios?

Semi-supervised learning (SSL) (Yang et al. 2022) provides an effective approach to leverage unlabeled data for improved model performance. The main idea is to estimate pseudo-labels for unlabeled data, enhancing model training alongside data augmentation methods. Thus far, researchers have developed SSL methods for centralized federated learning scenarios (Liang et al. 2022; Li, Li, and Wang 2023), where a central server facilitates the sharing of supervised information among clients. However, applying these methods to DFL is challenging due to the lack of central coordination. To the best of our knowledge, no SSL method has been specifically designed for DFL with diverse data sources and non-IID data distributions.

Designing a practical and effective semi-supervised DFL method presents several challenges introduced by limited labels in non-IID settings. First, accurately estimating pseudo-labels is crucial in SSL, but non-IID datasets make unbiased predictions challenging for a client's local model in DFL. Second, relying only on labeled and pseudo-labeled data may be insufficient for model training particularly for clients with limited labeled data, while unbiased data generation in highly non-IID DFL scenarios is challenging. In this paper, we propose a semi-supervised learning paradigm for DFL (SemiDFL) to deal with the above challenges, by establishing consensus in both model and data spaces.

Specifically, we propose a neighborhood pseudo-labeling method that introduces neighborhood classifiers for estimat-

---

[*] Equal Contribution.

[†] Bo Liu is the corresponding author.

ing pseudo-labels and uses neighborhood-qualified pseudo-label numbers to update the filtering threshold. This approach effectively improves the quality of pseudo-labeling and qualified pseudo-labeled data filtering, thereby enhancing classifier model training. We then design a consensus-based diffusion model for each client to generate synthesized data with a similar data distribution for further data MixUp operation combined with labeled and pseudo-labeled data. This forms a consensus data space among clients and therefore benefits the model training process and alleviates the non-IID issue. Instead of using average aggregation in the global consensus process, we design an adaptive aggregation method based on each classifier's accuracy on synthesized data generated by its diffusion model. This approach enhances model aggregation efficiency, forming a more effective consensus model space for both classifier and diffusion, which alleviates data source divergence and improves overall model performance.

Our paper mainly makes the following contributions:

- **SemiDFL Paradigm**: To our best knowledge, SemiDFL is the first semi-supervised DFL paradigm that is effective for clients with diverse data sources (labeled, unlabeled, and both) in highly non-IID settings. We innovatively propose consensus model and data spaces in SemiDFL to address the challenges of limited labels and non-IID data in semi-supervised DFL scenarios.
- **Consensus data space**: We design neighborhood pseudo labeling to improve the pseudo labeling quality of each client by combining its neighborhood information, and utilize the consensus-based DFL rule to train a unified diffusion model to generate synthesized data for further data MixUp. This helps to build a consensus data space among all clients to boost the classifier training.
- **Consensus model space**: We design an adaptive consensus mechanism based on each classifier model's performance on generated data to dynamically aggregate neighborhood diffusion and classifier models, circumventing the need of an extra shared dataset for model performance evaluation. This helps establish a better consensus model space across diverse clients.
- **Extensive Experiments**: We comprehensively evaluate SemiDFL through extensive experiments on different datasets, models, SSL settings, and non-IID degrees. The results verified SemiDFL's superior performance against existing methods.

Preliminaries and Related Work

Decentralized Federated Learning

Consider a DFL system with a set $\mathcal{N} = \{1, 2, \ldots, N\}$ of clients, where each client i has a general local model θ_i and a private dataset $\mathcal{X}_i$ containing $|\mathcal{X}_i|$ data samples. The goal of DFL is to minimize the total loss across all N clients:

$$\min_{\Theta} \frac{1}{N} \sum_{i=1}^{N} l(\theta_i; \mathcal{X}_i), \tag{1}$$

where $\Theta = \{\theta_i, i \in \mathcal{N}\}$ is the set of all local θ_i, $l(\theta_i; \mathcal{X}_i)$ represents the local loss function of client i. DFL typically runs over T rounds, with the following procedures executed sequentially in each round t:

- *Local training:* Each client i independently trains its local model on its dataset $\mathcal{X}_i$. The update is given by:

$$\theta_i^{t+1/2} = \theta_i{}^t - \eta \nabla \theta_i^t, \tag{2}$$

  where $\theta_i^{t+1/2}$ is the locally optimized model, η is the learning rate, and $\nabla \theta_i^t$ is the gradient. Notably, it can be multiple local training iterations in each round t.
- *Global consensus:* Each client i exchanges its local optimized model with its connected neighbors, and then aggregates the updates using a consensus mechanism:

$$\theta_i^{t+1} = \sum_{j \in \mathcal{G}_i} w_{ij} \theta_j^{t+1/2}, \tag{3}$$

  where $\mathcal{G}_i$ denotes the sub-graph centered on client i (including client i and its neighbors), w_{ij} is the aggregation weight of client j on client i, θ_i^{t+1} is the globally updated model for client i in round $t+1$.

Semi-supervised Learning Objective in DFL

We consider a practical SSL scenario with three types of clients based on their various data sources.

- *Labeled Client (L-client):* clients with only a few labeled data samples.
- *Unlabeled Client (U-client):* clients with only unlabeled data samples.
- *Mixed Client (M-client):* clients with both a few labeled data samples and many unlabeled samples.

We consider a system consisting of L-client set $\mathcal{N}_L$, U-client set $\mathcal{N}_U$, and M-client set $\mathcal{N}_M$. The aim of semi-supervised learning in DFL system is to jointly minimize the objectives of all clients as follows:

$$\min_{\Theta} \frac{1}{N} \Big(\sum_{i \in \mathcal{N}_L} l(\mathcal{L}_i; \theta_i) + \sum_{j \in \mathcal{N}_U} l(\mathcal{U}_j; \theta_j) + \sum_{k \in \mathcal{N}_M} l(\mathcal{M}_k; \theta_k) \Big), \tag{4}$$

where $\mathcal{L}_i, \mathcal{U}_i$ and $\mathcal{M}_i$ represent the datasets of L-, U- and M-clients, respectively. For $\mathcal{M}_i$, we further denote its labeled and unlabeled data as $\mathcal{M}_i'$ and $\mathcal{M}_i''$, respectively.

Related Works

Semi-supervised learning (SSL) (Yang et al. 2022) aims to leverage unlabeled data to enhance machine learning model performance, especially when unlabeled data predominates. A classical SSL method is pseudo-labeling (Lee et al. 2013), an entropy minimization approach (Grandvalet and Bengio 2004) that estimates a data sample's label based on model predictions. Pseudo-labeling is often combined with data augmentation techniques (e.g., MixUp (Zhang et al. 2017), MixMatch (Berthelot et al. 2019), FixMatch (Sohn et al. 2020)) to boost SSL performance. The main idea behind data augmentation is to expand the training dataset by interpolating data samples. Recent work (You et al. 2024) proposes using a diffusion model to generate synthesized data, further improving data augmentation in SSL.

Existing federated learning approaches primarily focus on supervised learning with fully labeled datasets, which may be impractical in many real-world scenarios, as noted in (Jin et al. 2020). One research direction assumes a labeled server with unlabeled clients (Diao, Ding, and Tarokh 2022, 2021; Albaseer et al. 2020). In (Jeong et al. 2021), model parameters are divided between servers with labeled data and clients with unlabeled data. The study (Zhang et al. 2021b) suggests training and aggregating a server model with labeled data to guide parallel client models with unlabeled data.

Another research direction assumes that some clients have labeled data. The work (Fan, Hu, and Huang 2022) proposes using a shared GAN model to generate synthesized data, creating a unified data space to enhance federated SSL performance. RSCFed (Liang et al. 2022) tackles the challenges of federated semi-supervised learning in non-iid settings by employing random sub-sampling and distance-reweighted model aggregation. CBAFed (Li, Li, and Wang 2023) uses a fixed pseudo-labeling strategy to prevent catastrophic forgetting and designs class-balanced adaptive thresholds based on local data distributions to address SSL challenges in the federated learning setting.

However, without a central coordinator server, it is challenging to apply SSL methods for FL to DFL scenarios with varying data sources and non-IID data distribution. To our knowledge, there is currently no SSL method specifically tailored for DFL that addresses the challenges of diverse data sources and non-IID data distributions.

Methodology

Overview of SemiDFL

SemiDFL aims to establish consensus in both model and data spaces among connected clients, addressing the challenge of limited labels in semi-supervised DFL without data sharing. It achieves consensus model space through global consensus on classifiers and diffusion models (presented by ϕ_i and ψ_i respectively, $i \in \mathcal{N}$) during the DFL training process. All consensus-based diffusion models generate synthesized data following a similar distribution, which is then mixed with labeled and pseudo-labeled data to form a consensus data space. The framework of SemiDFL is illustrated in Figure 1, with key components described below:

- **Neighborhood Pseudo-Labeling**: After training local classifier model $\phi_i, i \in \mathcal{N}$, client i first estimates pseudo-labels of its unlabeled samples (if it has) using local model ϕ_i and then filters unlabeled samples with qualified pseudo-label. Notably, neighborhood classifiers are introduced to enhance the pseudo-labeling process, and neighborhood-qualified pseudo-label numbers are used to adaptively update the filtering threshold.
- **Consensus MixUp**: Following pseudo-labeling, each client i trains a diffusion model ψ_i using labeled and/or pseudo-labeled samples. Notably, all diffusion models $\psi_i, i \in \mathcal{N}$ are aggregated based on a consensus mechanism, leading to a unified diffusion model. This makes synthesized data $\mathcal{D}_i, i \in \mathcal{N}$ generated from diffusion models $\psi_i, i \in \mathcal{N}$ follows a similar distribution (consensus data space) for subsequent MixUp.
- **Adaptive Aggregation**: Instead of using constant weights for classifier and diffusion models aggregation to form consensus model space, each client i adaptively aggregates its neighborhood models w_{ij} based on classifier performance. Each client i samples an extra small dataset from $\mathcal{D}_i, i \in \mathcal{N}$ (share a similar distribution) to evaluate its classifier. This not only ensures that all local models are evaluated on a shared dataset but also helps to circumvent data island problems in DFL.

Neighborhood Pseudo-Labeling

Pseudo-Labeling For the n-th indexed unlabeled data sample $u_i^n \in \mathcal{U}_i$ or $u_i^n \in \mathcal{M}_i''$ of client i, we estimate its pseudo-label using the client's classifier model ϕ_i, combined with label-invariant data augmentation and a label sharpening method (Berthelot et al. 2019). We first compute the label prediction $\bar{p}_n^i \in \mathbb{R}^{1\times C}$ (C is the number of categories) for sample u_i^n as follows:

$$\bar{p}_i^n = \frac{1}{K}\sum_{k=1}^{K} F(u_{i,k}^n; \phi_i), \tag{5}$$

where K is the number of variants for data augmentation, such as image rotation and cropping, $F(u_{i,k}^n; \phi_i)$ is the classifier with $u_{i,k}^n$ being the k-th augmented variant of sample u_i^n.

We then estimate the pseudo-label of u_i^n using label sharpening. The probability $\hat{p}_i^{n,c}$ that u_i^n is predicted as class c is given by:

$$\hat{p}_i^{n,c} = \frac{(\bar{p}_i^{n,c})^Z}{\sum_{c'=1}^{C}(\bar{p}_i^{n,c'})^Z}, \tag{6}$$

where $\bar{p}_i^{n,c}$ is the probability of class c in $\bar{p}_i^n$, and $Z > 1$ amplifies the dominating classes.

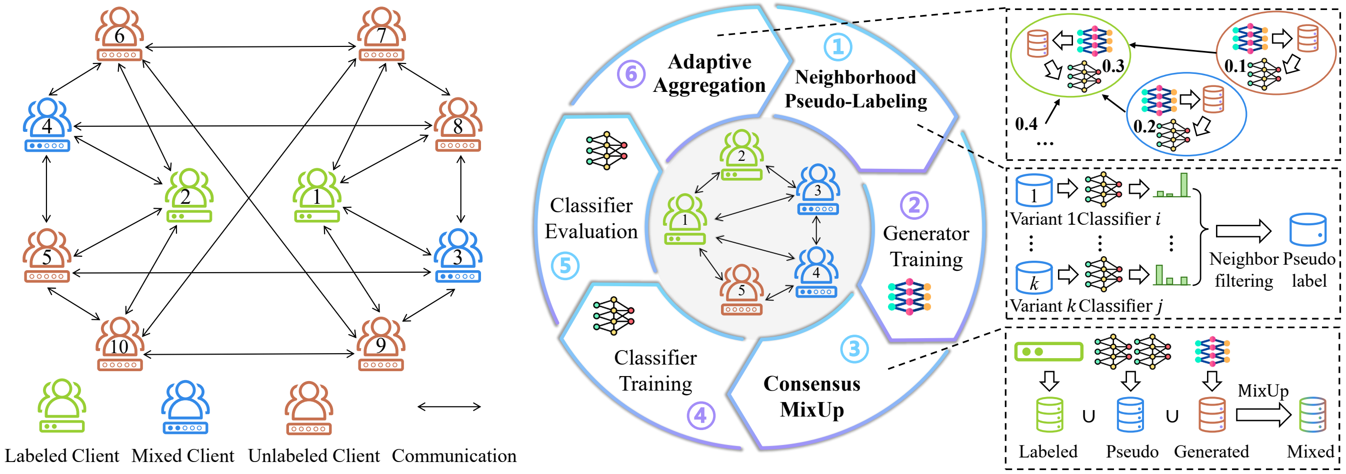
Neighborhood Pseudo-Labeling (NPL) In the challenging SemiDFL scenario with non-IID data, traditional pseudo-labeling methods suffer from high divergence among clients and noisy pseudo-labels, making them inadequate for providing high-quality supervision independently. We propose leveraging neighborhood information to address these challenges.

On the one hand, we introduce neighborhood classifiers to predict the label of a sample u_i^n's variant in client i. This design combines client i and its neighborhood classifiers' label predictions, thereby enhancing the robustness of pseudo-label prediction and reducing instability in highly non-IID settings. Specifically, we modify the label prediction as:

$$\bar{p}_i^n = \frac{1}{K}(F(u_{i,1}^n; \phi_i) + \sum_{k=2}^{K} F(u_{i,k}^n; \phi_k)), \tag{7}$$

where ϕ_k is a randomly selected model from the neighborhood set $\mathcal{G}_i$ of client i, i.e., $\{\phi_j, j \in \mathcal{G}_i\}$, for each data variant $u_{i,k}^n$.

On the other hand, we design a neighborhood adaptive class-wise threshold to filter noisy pseudo-label predictions, inspired by the dynamic threshold determination in (Li, Li,



(a) Communication topology

(b) Schematic diagram of SemiDFL

Figure 1: Framework of SemiDFL. (a) is an example of a decentralized communication topology employed in our experiments; (b) illustrates the overall process of the proposed SemiDFL, which consists of six main steps as indexed. Among these steps, steps 2, 4, and 5 are general ideas, while steps 1, 3 and 6 are the main contributions of our work. The detailed working flows are illustrated on the right of (b).

and Wang 2023). First, we calculate the number of samples with pseudo-label prediction probability above a manually defined threshold τ, given by:

$$\sigma_i^{t,c} = \sum_{n=1}^{|\mathcal{U}i|} \mathbb{1}\left(\max_c(\hat{p}_i^{n,c}) > \tau\right) \cdot \mathbb{1}\left(\arg\max_c(\hat{p}_i^{n,c}) = c\right), \quad (8)$$

where $\sigma_i^{t,c}$ is the number of samples of all the unlabeled data in current client i with pseudo-label prediction probability above the threshold in class c and $\mathbb{1}(\cdot)$ is an indicator, taking 1 if the condition in the parentheses is true, and 0 otherwise.

Then, the threshold is adaptively updated by:

$$\tau_i^{t,c} = \frac{\sigma_i^{t,c}}{\max_i(\max_c(\sigma_i^{t,c}), i \in \mathcal{G}_i)} \tau, \quad (9)$$

where $\tau_i^{t,c}$ is the neighborhood adaptive threshold. This helps balance the number of pseudo-label samples, reducing the negative impact of non-IID data.

Consensus MixUp

Local MixUp (L-MixUp) After filtering unlabeled samples with qualified pseudo-labels for client i, we generate mixed samples using MixUp based on the union set of labeled dataset $\mathcal{L}_i$ and pseudo-labeled samples $\mathcal{P}_i$ for local classifier model training. Suppose (x_m, y_m) and (x_n, y_n) are two m- and n-th indexed data samples in the union set $\mathcal{L}_i \cup \mathcal{P}_i$. The MixUp operation produces:

$$\begin{aligned} x' &= \lambda x_m + (1-\lambda) x_m, \\ y' &= \lambda y_m + (1-\lambda) y_n, \end{aligned} \quad (10)$$

where λ is a hyper-parameter that determines the mixing ratio, (x', y') is the mixed data sample.

Consensus MixUp (C-MixUp) L-MixUp leads to suboptimal performance in the DFL scenarios with few and non-IID labeled data because data sharing is not allowed among agents. For this issue, we propose the consensus MixUp (C-MixUp) which facilitates data imputation across all agents without raw data sharing.

C-MixUp employs a generative learning manner to form a consensus data space, enabling the generation and mixing of data samples without data sharing. For each client i, we adopt a diffusion model to generate synthesized data (Sohl-Dickstein et al. 2015; Ho, Jain, and Abbeel 2020). In the training phase, we first obtain the noisy latent X_h by progressively adding noise to input data X_0, where h denotes a randomly sampled diffusion timestep. Then, We train a diffusion model $D(\cdot;\psi_i)$ to predict the noise ϵ added on X_h by minimizing the following objective $l'(\cdot;\psi_i)$:

$$l'(\cdot;\psi_i) = \mathbb{E}_{h, X_0, \epsilon\sim\mathcal{N}(0,\mathrm{I})}[\|D(X_h, c, h; \psi_i) - \epsilon\|_2^2] \quad (11)$$

where ψ_i is parameter of the diffusion model, x is a data sample with c being its class. Notably, the X_0 could come from the labeled or unlabeled dataset, and c maybe its real class or generated pseudo-label.

In the sampling phase, pseudo-images are generated via deterministic sampling (Song, Meng, and Ermon 2020; Lu et al. 2022) combined with the classifier-free method (Ho and Salimans 2022) from the trained diffusion model. We denote this generated dataset by diffusion model as $\mathcal{D}_i$. Notably, the local diffusion model ψ_i is globally updated through a consensus mechanism during the DFL training process, as shown in (3). This ensures that all local ψ_i converge to a unified model, thus all generated datasets $\mathcal{D}_i, i \in \mathcal{N}$ follow a similar data distribution.

The generated data from $\mathcal{D}_i$ is then combined with $\mathcal{X}_i$ and $\mathcal{P}_i$ to produce more mixed samples for enhancing local training. Specifically, we create a new union set $\mathcal{L}_i \cup \mathcal{P}_i \cup \mathcal{D}_i$ to improve the MixUp process. This forms a consensus data space.

Adaptive Aggregation In the vanilla DFL setting, each agent's aggregation weight is fixed, based on the communication topology and consensus strategy. This unchanging influence on neighbors is inefficient in the DFL-Semi scenario. Ideally, agents with better performance should exert more influence, i.e., have larger aggregation weights. However, assuming a shared test dataset to evaluate each client's performance is impractical due to privacy constraints.

Nevertheless, since all datasets $\mathcal{D}_i, i \in \mathcal{G}_i$ share a similar distribution, making them ideal for performance evaluation. To reduce computing cost, each client randomly samples an tiny validation dataset (100 samples in our experiments) $\hat{\mathcal{D}}_i^t$ from $\mathcal{D}_i$ to evaluate its performance for determining the aggregation weights of neighbors at the t-th round.

We use a_i to denote the accuracy of model ϕ_i on the validation dataset $\hat{\mathcal{D}}_i^t$. The adaptive weight is then obtained by

$$w_{ij} = \frac{\exp(a_j - \bar{a}_i)}{\sum_{j \in \mathcal{G}_i} \exp(a_j - \bar{a}_i)}, \tag{12}$$

where $\bar{a}_i = \frac{1}{|\mathcal{G}_i|} \sum_{j \in \mathcal{G}_i} a_j$ is the averaged accuracy in the sub-graph $\mathcal{G}_i$ with $|\mathcal{G}_i|$ being the number of clients in the sub-graph.

Based on the above three main components, we proposed SemiDFL as detailed in **Algorithm** 1 [1].

Experiments

Experimental Setup

We use the decentralized communication topology in Figure 1(a) as an example to evaluate SemiDFL on different datasets, various labeled data ratios, and non-IID degrees. More detailed experimental settings and parameters can be found in the supplementary material.

Datasets and Models We evaluate SemiDFL on MNIST (LeCun, Cortes, and Burges 1998) and Fashion-MNIST (Xiao, Rasul, and Vollgraf 2017) using Convolutional Neural Network (CNN), and on CIFAR-10 (Krizhevsky, Nair, and Hinton 2010) using ResNet-18.

Hyper-Parameters We adopt the same method in (Hsu, Qi, and Brown 2019; Lin et al. 2020) to simulate different non-IID data distribution degrees. Specifically, the non-IID degree is captured by a sample allocation probability α, with smaller α indicating a higher non-IID degree. The percentage of total labeled data in the union of all clients' data is denoted by the labeled data ratio r. We train models for 500 global rounds on all datasets. In each global training round, each client performs E (25 for MNIST, 50 for Fashion-MNIST and CIFAR-10) iterations of local training via mini-batch SGD with a batch size of $B = 10$. Other hyper-parameters during local model training are inherited from the default settings of Adam (Kingma and Ba 2014) and for all datasets, we use a learning rate of 0.05. All experiments are conducted using PyTorch 2.0 on a machine with 2 RTX 4090 GPUs.

Algorithm 1: SemiDFL

```
Inputs: Client number $N$, L-client set $\mathcal{N}_L$, U-client
  set $\mathcal{N}_U$, M-client set $\mathcal{N}_M$, communication topology
  $\mathcal{G}$, global training rounds $T$, diffusion warm-up
  round $R$.
Outputs: Optimal classifier $\phi_i$ for client $i \in \mathcal{N}$.
Initialization: Local classifier model $\phi_i^0$ and local
  diffusion model $\psi_i^0$ for client $i \in \mathcal{N}$.
for round $t \in \{1, 2, \dots, T\}$ do
    for client $i \in \mathcal{N}$ in parallel do
        if $i \in \mathcal{N}_L$ then $Y_i = \mathcal{L}_i$; $Z_i = \emptyset$.
          else if $i \in \mathcal{N}_U$ then $Y_i = \emptyset$; $Z_i = \mathcal{U}_i$.
          else if $i \in \mathcal{N}_M$ then $Y_i = \mathcal{M}_i'$; $Z_i = \mathcal{M}_i''$.
        end
        $\mathcal{P}_i^t \leftarrow$ NPL($Z_i$,$\{\phi_j^t, j \in \mathcal{G}_i\}$) by (6)-(9).
        $\psi_i^{t+\frac{1}{2}} \leftarrow$ Train diffusion on $Y_i \cup \mathcal{P}_i^t$ by (11), (2).
        if $t \geq R$ then
            $\mathcal{D}_i^t \leftarrow$ Generate data from diffusion $\psi_i^{t+\frac{1}{2}}$.
        end
        $S_i^t \leftarrow$ C-MixUp($Y_i \cup \mathcal{P}_i^t \cup \mathcal{D}_i^t, \lambda$).
        $\phi_i^{t+\frac{1}{2}} \leftarrow$ Train classifier on $S_i^t$ by (1), (2).
        $\hat{\mathcal{D}}_i^t \leftarrow$ Small validation dataset from $\mathcal{D}_i^t$.
        $a_i \leftarrow$ Model evaluation of $\phi_i^{t+1/2}$ on $\hat{\mathcal{D}}_i^t$.
    end
    for client $i \in \mathcal{N}$ in parallel do
        $w_{ij}^t, j \in \mathcal{G}_i \leftarrow$ Compute weight by (12).
        $\phi_i^{t+1} \leftarrow$ Consensus update $\phi_i^{t+1/2}$ by (3),
        $\psi_i^{t+1} \leftarrow$ Consensus update $\psi_i^{t+1/2}$ by (3).
    end
end
```

Baseline methods We use the following SSL methods as baselines by adapting them to fit the DFL scenario.

- **DFL upper bound (DFL-UB)**: This represents the DFL upper bound, where the model is trained assuming all data are labeled.
- **DFL Lower Bound (DFL-LB)**: This represents the DFL lower bound, where only the partially labeled data is available for model training.
- **MixMatch(Berthelot et al. 2019)**: MixMatch assigns low-entropy labels to the augmented unlabeled data samples and blends them with labeled data using MixUp.
- **FlexMatch (Zhang et al. 2021a)**: FlexMatch improves Fixmatch (Sohn et al. 2020) by flexibly adjusting class-specific thresholds at each round to select informative pseudo-labeled data.
- **CBAFed (Li, Li, and Wang 2023)**: CBAFed addresses catastrophic forgetting with fixed pseudo-labeling, uses class-balanced adaptive thresholds for training balance,

---

[1] The variable Y_i and Z_i are to denote client i's label data and unlabeled data, respectively. The symbols $\mathcal{M}_i'$ and $\mathcal{M}_i''$ represent the labeled and unlabeled data of M-client i, respectively.

Setting / Method	non-IID Degree	MNIST		Fashion-MNIST		CIFAR-10	
		$r = 0.5\%$	$r = 0.1\%$	$r = 0.5\%$	$r = 0.1\%$	$r = 5\%$	$r = 1\%$
DFL-UB	$\alpha = 100$	95.76 ± 0.10		86.12 ± 0.14		86.11 ± 0.08	
	$\alpha = 0.1$	94.27 ± 0.76		73.64 ± 3.58		58.75 ± 4.12	
DFL-LB	$\alpha = 100$	89.24 ± 0.06	68.98 ± 0.02	65.16 ± 1.55	48.53 ± 1.76	55.96 ± 0.11	41.30 ± 0.03
	$\alpha = 0.1$	82.36 ± 1.18	66.49 ± 0.05	49.08 ± 5.46	——	16.96 ± 0.27	——
MixMatch	$\alpha = 100$	86.89 ± 0.69	61.13 ± 18.27	65.97 ± 1.65	35.82 ± 1.75	65.54 ± 0.23	48.16 ± 0.16
	$\alpha = 0.1$	63.69 ± 14.62	39.40 ± 10.90	54.52 ± 3.14	42.46 ± 4.28	44.49 ± 1.15	30.53 ± 1.37
FlexMatch	$\alpha = 100$	91.85 ± 0.23	71.84 ± 1.03	69.33 ± 1.12	52.40 ± 0.26	56.90 ± 0.18	37.91 ± 0.20
	$\alpha = 0.1$	81.38 ± 2.58	55.96 ± 2.31	57.63 ± 2.37	40.03 ± 4.05	46.09 ± 0.50	——
CBAFed	$\alpha = 100$	69.93 ± 1.61	40.67 ± 0.44	55.54 ± 1.16	27.99 ± 0.22	42.29 ± 0.91	26.18 ± 0.23
	$\alpha = 0.1$	70.65 ± 4.64	37.79 ± 2.48	47.39 ± 4.40	——	31.93 ± 1.13	21.50 ± 0.89
SemiDFL	$\alpha = 100$	$\mathbf{94.46 \pm 0.06}$	$\mathbf{90.88 \pm 0.07}$	$\mathbf{75.36 \pm 0.27}$	$\mathbf{60.27 \pm 0.16}$	$\mathbf{69.40 \pm 0.20}$	$\mathbf{49.79 \pm 0.18}$
	$\alpha = 0.1$	$\mathbf{88.69 \pm 0.09}$	$\mathbf{70.66 \pm 0.50}$	$\mathbf{71.49 \pm 0.14}$	$\mathbf{49.40 \pm 0.71}$	$\mathbf{49.63 \pm 0.61}$	$\mathbf{40.28 \pm 0.89}$

Table 1: Averaged accuracy ± Standard deviation of all clients on MNIST, Fashion-MNIST, and CIFAR-10 datasets. (DFL-UB assumes all data are labeled, so it remains unchanged with varying r. The symbol "——" denotes a non-convergent result.)

and employs residual weight connections for optimized model aggregation.

Comparison Study

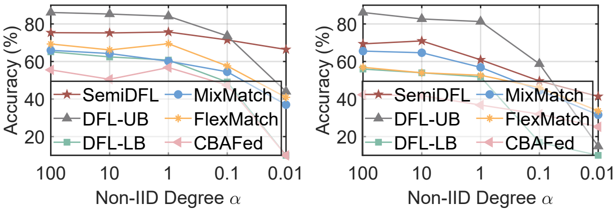
Table 1 presents the quantitative results of our SemiDFL alongside the state-of-the-art baselines on three datasets. Our proposed SemiDFL consistently outperforms all baselines (except DFL-UB) across different non-IID degrees α and various labeled data ratios r. Notably, SemiDFL achieves higher averaged accuracy and a smaller standard deviation of all clients compared with other SSL methods, benefiting from its established consensus model and data space. This demonstrates the effectiveness and superiority of SemiDFL in handling diverse semi-supervised DFL tasks with varying data settings.

Robustness on various non-IID degrees Figure 2 illustrates model performance on the Fashion-MNIST and CIFAR-10 datasets across various non-IID degrees (with smaller α indicating higher non-IID degree). The results show that SemiDFL consistently outperforms other baselines (except DFL-UB) with higher accuracy under different non-IID degrees. Although almost all methods experience performance declines as the non-IID degree increases, SemiDFL exhibits a smaller decrease in accuracy in highly non-IID scenarios on easy tasks (e.g. Fashion-MNIST dataset), demonstrating its robustness over non-IID data distribution among clients.

Robustness on various labeled data ratios Figure 3 demonstrates the accuracy of different models across various labeled data ratios on two datasets. Our method consistently outperforms other baselines (except DFL-UB) and exhibits a smaller performance decrease with a low labeled data ratio, exhibiting its robustness on various labeled data ratios.

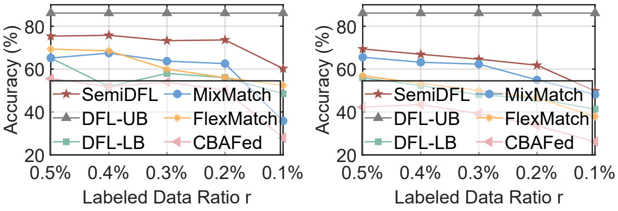
Ablation Study

Neighborhood Pseudo-Labeling To further verify the effectiveness of neighborhood pseudo-labeling, we compare the following three variants of pseudo-labeling methods.



(a) Fashion-MNIST ($r = 0.5\%$). (b) CIFAR-10 ($r = 5\%$).

Figure 2: Accuracy versus non-IID degree.



(a) Fashion-MNIST. (b) CIFAR-10.

Figure 3: Accuracy versus labeled data ratio ($\alpha = 100$).

- **Vanilla Pseudo-Labeling (Vanilla PL)**: each client takes the vanilla pseudo-labeling method without neighborhood classifier and filtering threshold.
- **Adaptive Pseudo-Labeling (APL)**: each client operates a pseudo-labeling method without neighborhood information, while the filtering threshold is adaptively updated based on the local number of qualified pseudo-labeled data.
- **Neighborhood Pseudo-Labeling (NPL, Ours)**: each client introduces neighborhood classifiers to enhance the pseudo-labeling process and uses neighborhood-qualified pseudo-label numbers to adaptively update the filtering threshold.

Method Setting	Vanilla PL	APL	NPL (Ours)
$\alpha = 100$ $r = 5\%$	40.51 ± 13.74	47.40 ± 9.83	$\mathbf{69.40 \pm 0.20}$
$\alpha = 100$ $r = 1\%$	32.13 ± 0.67	34.88 ± 0.89	$\mathbf{49.79 \pm 0.18}$
$\alpha = 0.1$ $r = 5\%$	24.51 ± 3.98	31.14 ± 5.36	$\mathbf{49.63 \pm 0.61}$
$\alpha = 0.1$ $r = 1\%$	14.95 ± 2.22	30.89 ± 1.03	$\mathbf{40.28 \pm 0.89}$

Table 2: Ablation of neighborhood pseudo-labeling on CIFAR-10.

Table 2 shows that NPL consistently outperforms both Vanilla PL and Adaptive APL across all settings. While APL improves Vanilla PL by incorporating an adaptive filtering threshold, it still lags in performance within the DFL scenario due to dynamic and noisy supervision, especially in highly non-IID cases. NPL significantly enhances model performance by leveraging neighborhood information and the number of qualified pseudo-labels in each round to refine supervision. This verifies the effectiveness and superiority of the proposed neighborhood pseudo-labeling method.

Consensus MixUp To demonstrate the efficacy of the proposed C-MixUp method, we compare the following three variants. Notably, a client's labeled and pseudo-labeled data can be an empty set if it lacks this data source.

- **Local MixUp (L-MixUp)**: Each client performs local MixUp using labeled and pseudo-labeled data to generate mixed data for local classifier training.
- **C-MixUp with GAN (w/ GAN)**: Each client performs C-MixUp leveraging labeled data, pseudo-labeled data, and synthesized data generated by a consensus-based GAN model (Odena, Olah, and Shlens 2017) to create mixed data for local classifier training.
- **C-MixUp with Diffusion (w/ Diffusion, Ours)**: Each client performs C-MixUp using labeled data, pseudo-labeled data, and synthesized data generated by a consensus-based diffusion model to produce mixed data for local classifier training.

Table 3 shows that our proposed C-MixUp method consistently outperforms all three variants. Additionally, both C-MixUp with diffusion and GAN achieve higher accuracy than L-MixUp, particularly in highly non-iid scenarios (small α). This verifies the efficacy of the consensus data space designed by C-MixUp in semi-supervised DFL tasks.

Adaptive Aggregation We examine the efficacy of adaptive aggregation by comparison with its three variants:

- **Constant weight**: Clients in a sub-graph use a constant weight to aggregate their classifier and diffusion models.
- **Adaptive on test dataset (AdaTest)**: Clients aggregate models based on adaptive weights determined by their accuracy on an extra test dataset. For this ablation study only, we assume clients have access to an extra shared test dataset.

Method Setting	L-MixUp	w/ GAN	w/ Diffusion (Ours)
$\alpha = 100$ $r = 0.5\%$	93.69 ± 0.09	92.17 ± 0.20	$\mathbf{94.46 \pm 0.06}$
$\alpha = 100$ $r = 0.1\%$	76.30 ± 0.54	90.29 ± 0.10	$\mathbf{90.88 \pm 0.07}$
$\alpha = 0.1$ $r = 0.5\%$	70.75 ± 9.39	86.80 ± 0.47	$\mathbf{88.69 \pm 0.09}$
$\alpha = 0.1$ $r = 0.1\%$	49.13 ± 4.93	69.32 ± 0.89	$\mathbf{70.66 \pm 0.50}$

Table 3: Ablation of Consensus-MixUp on MNIST.

Method Setting	Constant	AdaTest	AdaGen (Ours)
$\alpha = 100$ $r = 0.5\%$	75.18 ± 0.22	$\mathbf{75.80 \pm 0.12}$	75.36 ± 0.27
$\alpha = 100$ $r = 0.1\%$	56.12 ± 0.11	59.03 ± 0.13	$\mathbf{60.27 \pm 0.16}$
$\alpha = 0.1$ $r = 0.5\%$	70.06 ± 0.34	70.46 ± 0.78	$\mathbf{71.49 \pm 0.14}$
$\alpha = 0.1$ $r = 1\%$	49.11 ± 0.56	$\mathbf{51.38 \pm 0.64}$	49.40 ± 0.71

Table 4: Ablation of adaptive aggregation on Fashion-MNIST.

- **Adaptive on generated dataset (AdaGen, Ours)**: All clients aggregate models based on adaptive weights determined by their accuracy on the dataset generated by diffusion models.

The results in Table 4 demonstrate that the proposed AdaGen outperforms constant weight aggregation and shows comparable performance to AdaTest, despite not having access to an extra shared test dataset. This enables effective performance evaluation while preserving data privacy.

Conclusion

This paper proposes SemiDFL, the first semi-supervised DFL paradigm through constructing consensus data and model spaces among clients to tackle the challenges of limited labels and highly non-IID data distributions in DFL. SemiDFL utilizes neighborhood information to enhance the estimation and filtering of pseudo-labels for unlabeled samples, improving both the quality and robustness of pseudo-labeling. Additionally, a consensus-based diffusion model generates synthesized data with a similar distribution, facilitating MixUp and forming a consensus data space that mitigates non-IID issues in classifier training. We further design an adaptive aggregation strategy based on each client's performance to establish a more effective consensus model space, enhancing classifier performance. Extensive experimental evaluations demonstrate the superior performance of SemiDFL compared to existing semi-supervised learning methods in DFL scenarios.

Acknowledgements

This work is supported by National Natural Science Foundation of China (Grant No.62203309, 62401161), Guangdong Basic and Applied Basic Research Foundation (Grant No. 2024A1515011333, 2022A1515110056), Shenzhen Science and Technology Program (Grant No. RCBS20221008093312031), Longgang District Shenzhen's "Ten Action Plan"for Supporting Innovation Projects (Grant No. LGKCSDPT2024002, LGKCSDPT2024003), and the Shenzhen Institute of Artificial Intelligence and Robotics for Society.

References

Albaseer, A.; Ciftler, B. S.; Abdallah, M.; and Al-Fuqaha, A. 2020. Exploiting unlabeled data in smart cities using federated learning. *arXiv preprint arXiv:2001.04030*.

Beltrán, E. T. M.; Pérez, M. Q.; Sánchez, P. M. S.; Bernal, S. L.; Bovet, G.; Pérez, M. G.; Pérez, G. M.; and Celdrán, A. H. 2023. Decentralized federated learning: Fundamentals, state of the art, frameworks, trends, and challenges. *IEEE Communications Surveys & Tutorials*.

Berthelot, D.; Carlini, N.; Goodfellow, I.; Papernot, N.; Oliver, A.; and Raffel, C. A. 2019. Mixmatch: A holistic approach to semi-supervised learning. *Advances in neural information processing systems*, 32.

Diao, E.; Ding, J.; and Tarokh, V. 2021. SemiFL: Communication efficient semi-supervised federated learning with unlabeled clients. *arXiv preprint arXiv:2106.01432*, 3.

Diao, E.; Ding, J.; and Tarokh, V. 2022. Semifl: Semi-supervised federated learning for unlabeled clients with alternate training. *Advances in Neural Information Processing Systems*, 35: 17871–17884.

Fan, C.; Hu, J.; and Huang, J. 2022. Private Semi-Supervised Federated Learning. In *IJCAI*, 2009–2015.

Grandvalet, Y.; and Bengio, Y. 2004. Semi-supervised learning by entropy minimization. *Advances in neural information processing systems*, 17.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Ho, J.; and Salimans, T. 2022. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*.

Hsu, T.-M. H.; Qi, H.; and Brown, M. 2019. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335*.

Jeong, W.; Yoon, J.; Yang, E.; and Hwang, S. J. 2021. Federated Semi-supervised Learning with Inter-client Consistency & Disjoint Learning. In *9th International Conference on Learning Representations, ICLR 2021*. International Conference on Learning Representations, ICLR.

Jin, Y.; Wei, X.; Liu, Y.; and Yang, Q. 2020. Towards utilizing unlabeled data in federated learning: A survey and prospective. *arXiv preprint arXiv:2002.11545*.

Kairouz, P.; McMahan, H. B.; Avent, B.; Bellet, A.; Bennis, M.; Bhagoji, A. N.; Bonawitz, K.; Charles, Z.; Cormode, G.; Cummings, R.; et al. 2021. Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14(1–2): 1–210.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Krizhevsky, A.; Nair, V.; and Hinton, G. 2010. Cifar-10 (canadian institute for advanced research). http://www.cs.toronto.edu/kriz/cifar.html. Accessed: 2024-05-06.

LeCun, Y.; Cortes, C.; and Burges, C. 1998. MNIST handwritten digit database, 1998. http://www.research.att.com/yann/ocr/mnist. Accessed: 2024-05-06.

Lee, D.-H.; et al. 2013. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, 896. Atlanta.

Li, M.; Li, Q.; and Wang, Y. 2023. Class balanced adaptive pseudo labeling for federated semi-supervised learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 16292–16301.

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

Lin, T.; Kong, L.; Stich, S. U.; and Jaggi, M. 2020. Ensemble distillation for robust model fusion in federated learning. *Advances in neural information processing systems*, 33: 2351–2363.

Liu, B.; and Ding, Z. 2021a. A consensus-based decentralized training algorithm for deep neural networks with communication compression. *Neurocomputing*, 440: 287–296.

Liu, B.; and Ding, Z. 2021b. Distributed Heuristic Adaptive Neural Networks With Variance Reduction in Switching Graphs. *IEEE Transactions on Cybernetics*, 51(7): 3836–3844.

Liu, B.; Ding, Z.; and Lv, C. 2020. Distributed Training for Multi-Layer Neural Networks by Consensus. *IEEE Transactions on Neural Networks and Learning Systems*, 31(5): 1771–1778.

Lu, C.; Zhou, Y.; Bao, F.; Chen, J.; Li, C.; and Zhu, J. 2022. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in Neural Information Processing Systems*, 35: 5775–5787.

McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; and y Arcas, B. A. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, 1273–1282. PMLR.

Odena, A.; Olah, C.; and Shlens, J. 2017. Conditional image synthesis with auxiliary classifier gans. In *International conference on machine learning*, 2642–2651. PMLR.

Sohl-Dickstein, J.; Weiss, E.; Maheswaranathan, N.; and Ganguli, S. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, 2256–2265. PMLR.

Sohn, K.; Berthelot, D.; Carlini, N.; Zhang, Z.; Zhang, H.; Raffel, C. A.; Cubuk, E. D.; Kurakin, A.; and Li, C.-L. 2020. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in neural information processing systems*, 33: 596–608.

Song, J.; Meng, C.; and Ermon, S. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Sun, P.; Liu, X.; Wang, Z.; and Liu, B. 2024. Byzantine-robust Decentralized Federated Learning via Dual-domain Clustering and Trust Bootstrapping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 24756–24765.

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

Yang, X.; Song, Z.; King, I.; and Xu, Z. 2022. A survey on deep semi-supervised learning. *IEEE Transactions on Knowledge and Data Engineering*, 35(9): 8934–8954.

You, Z.; Zhong, Y.; Bao, F.; Sun, J.; Li, C.; and Zhu, J. 2024. Diffusion models and semi-supervised learners benefit mutually with few labels. *Advances in Neural Information Processing Systems*, 36.

Zhang, B.; Wang, Y.; Hou, W.; Wu, H.; Wang, J.; Okumura, M.; and Shinozaki, T. 2021a. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in Neural Information Processing Systems*, 34: 18408–18419.

Zhang, H.; Cisse, M.; Dauphin, Y. N.; and Lopez-Paz, D. 2017. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*.

Zhang, Z.; Yang, Y.; Yao, Z.; Yan, Y.; Gonzalez, J. E.; Ramchandran, K.; and Mahoney, M. W. 2021b. Improving semi-supervised federated learning by reducing the gradient diversity of models. In *2021 IEEE International Conference on Big Data (Big Data)*, 1214–1225. IEEE.