Machine Intelligence Research
www.mi-research.net

21(3), June 2024, 526-537
DOI: 10.1007/s11633-023-1449-1

Overhead-free Noise-tolerant Federated Learning: A New Baseline

Shiyi Lin[1] Deming Zhai[1] Feilong Zhang[1]
Junjun Jiang[1] Xianming Liu[1] Xiangyang Ji[2]

[1] Department of Computer Science and Technology, Harbin Institute of Technology, Harbin 150000, China

[2] Department of Automation, Tsinghua University, Beijing 100084, China

Abstract: Federated learning (FL) is a promising decentralized machine learning approach that enables multiple distributed clients to train a model jointly while keeping their data private. However, in real-world scenarios, the supervised training data stored in local clients inevitably suffer from imperfect annotations, resulting in subjective, inconsistent and biased labels. These noisy labels can harm the collaborative aggregation process of FL by inducing inconsistent decision boundaries. Unfortunately, few attempts have been made towards noise-tolerant federated learning, with most of them relying on the strategy of transmitting overhead messages to assist noisy labels detection and correction, which increases the communication burden as well as privacy risks. In this paper, we propose a simple yet effective method for noise-tolerant FL based on the well-established co-training framework. Our method leverages the inherent discrepancy in the learning ability of the local and global models in FL, which can be regarded as two complementary views. By iteratively exchanging samples with their high confident predictions, the two models "teach each other" to suppress the influence of noisy labels. The proposed scheme enjoys the benefit of overhead cost-free and can serve as a robust and efficient baseline for noise-tolerant federated learning. Experimental results demonstrate that our method outperforms existing approaches, highlighting the superiority of our method.

Keywords: Federated learning, noise-label learning, privacy-preserving machine learning, edge intelligence, distributed machine learning.

Citation: S. Lin, D. Zhai, F. Zhang, J. Jiang, X. Liu, X. Ji. Overhead-free noise-tolerant federated learning: A new baseline. *Machine Intelligence Research*, vol.21, no.3, pp.526–537, 2024. http://doi.org/10.1007/s11633-023-1449-1

1 Introduction

With the success of data-driven deep neural networks (DNNs) in various applications, there are growing concerns for user privacy and data confidentiality. Federated learning (FL) offers a solution to this issue through its decentralized machine learning paradigm, where many distributed clients (e.g., mobile and edge devices, organizations, institutions) collaboratively train a model under the coordination of a central server[1]. In contrast to traditional centralized machine learning, FL shares a model between clients and a server instead of sharing the data itself, mitigating the systemic privacy risks and costs. FL is a promising approach to analyze and learn from data distributed among many owners without exposing that data. Recently, it has received significant interest from both research and applied perspectives[2, 3].

Federated learning typically follows a hub-and-spoke topology and involves two main stages that are iterated until the learning objective is achieved[1]. The two stages are: 1) Local training. Each client performs model training independently based on their own local data and the global model weights downloaded from the central server. The trained model weights are then uploaded back to the central server. 2) Global aggregation. The central server collects the aggregate of the local gradient updates and generates a new global model.

In real-world scenarios, federated learning encompasses a wide range of constraints and challenges. In this work, we specially consider the challenge of making federated learning more robust and efficient. Specifically, the local training is conducted in a supervised learning manner with large amounts of annotated training data stored in local clients. However, it is well-known that the real-word data usually suffers from imperfect annotations, resulting in subjective, inconsistent and biased labels. The so-called noisy labels would degrade the generalization performance of deep learning model significantly, since over-parameterized neural networks have enough capacity to memorize large-scale data with even completely random labels[4]. This problem is more serious in the FL

Research Article
Manuscript received on March 1, 2023; accepted on April 18, 2023; published online on January 12, 2024
Recommended by Associate Editor Min-Ling Zhang
Colored figures are available in the online version at https://link.springer.com/journal/11633

setting. The noisy labels would produce inconsistent decision boundaries and further divergent model weights among clients, which lead to individual models difficult to reach consensus and harm the collaborative aggregation process of the global model.

Learning with noisy labels (a.k.a., robust training) is a hot topic in traditional centralized machine learning, for which many strategies have been proposed[5–8]. Surprisingly, in the field of federated learning, there are very few attempts towards noise-tolerant federated learning[9–11]. Compared with its centralized counterpart, it is more challenging to detect and correct noisy labels in federated learning due to the data access mechanisms to protect privacy. The existing noise-tolerant FL methods commonly exploit the strategy of transmitting overhead information to assist noisy labels detection and correction. Yang et al.[10] propose to interchange class-wise centroids of local data on each device, which are aligned by the server every communication round, so as to help to form consistent decision boundaries among local models. Tam et al.[11] propose to send data quality and the amount of training data in each client to the server in each round, which are used to perform the weighted aggregation of the local models to update the global model. Chen et al.[9] propose to upload local models and the cross-entropy losses of the global model on local data of each client to the server where a benchmark set with convincing labels are maintained; then compute mutual cross-entropy between performance of the global model on the local datasets and that of the local model on the benchmark dataset, which is used for the subsequent reweighting procedure.

Privacy and communication efficiency are always first-order concerns in federated learning[12]. However, as outlined above, the current methods disclose the information of local data more or less. For instance, in [10], the server can easily infer which classes of data a local client owns according to the uploaded class centroids; in [11], the server can know the quality and size of data in a local client. Moreover, the uploaded overhead information in each round reduces the communication efficiency of the federated learning system. Considering the limitation of existing strategies, it is necessary to define a robust and efficient baseline for noise-tolerant federated learning, so as to promote the further research on this important but neglected topic in FL.

In this paper, we propose a simple yet effective method for noise-tolerant FL that integrates the disagreement-based training with FL, inspired by the classical co-training in semi-supervised learning[13]. As illustrated in Fig. 1, in each client, there are two diverse models available: the local model trained on local data of each client, and the downloaded global model trained by aggregating all local models. Therefore, it naturally offers us two learners trained from two different views, in contrast to its counterpart co-teaching[14] in centralized machine learning where two neural networks are specially maintained for this purpose.

The main contributions of this work are highlighted as follows:

1) We propose a simple yet effective scheme that can be easily integrated into the existing FL framework to handle mislabeled local data. Our method does not need to reveal any overhead information to the server, and thus enjoy benefits from both privacy preservation and communication efficiency.

2) Our proposed federated co-training scheme enjoys performance guarantee. As theoretically proven in [15], the sufficient and necessary condition for co-training to succeed is as long as there is large diversity of two

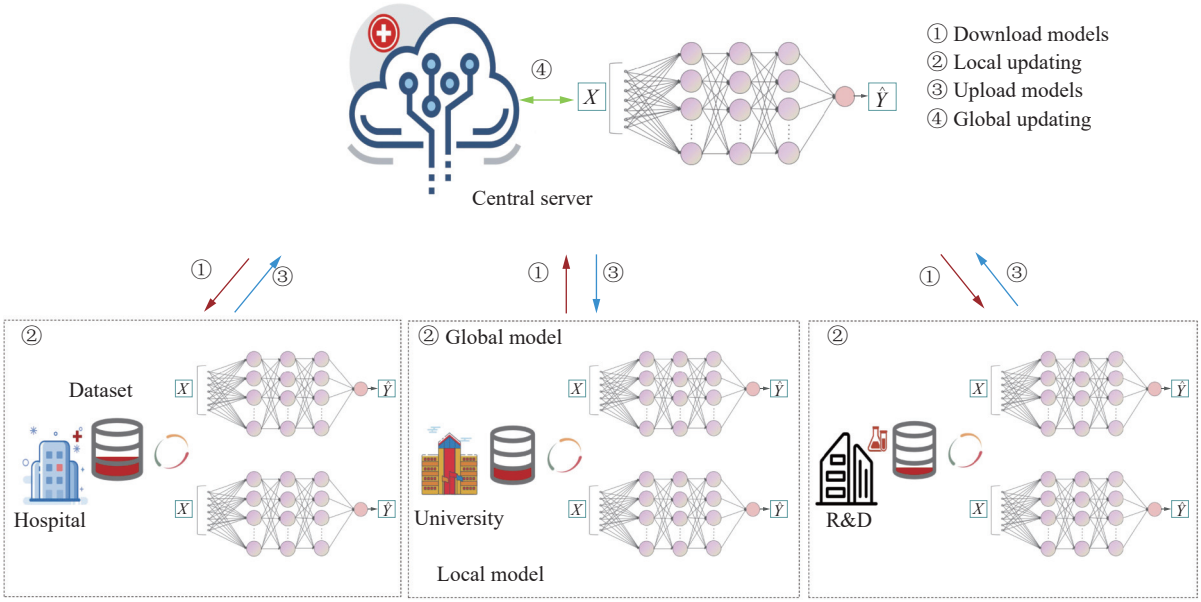


Fig. 1 Overview framework of our proposed noise-tolerant federated learning based on co-training

learners. In FL, the local models and the global model are inherently different, thus it can be easy to achieve this condition.

3) Extensive experimental results on both synthetic and real-world datasets demonstrate the superiority of our method over FedAvg[1], co-teaching[14] and state-of-the-art noise-tolerant FL scheme[10]. Our method can serve as a new baseline for noise-tolerant FL.

2 Related works

2.1 Federated learning

FedAvg[1] or local SGD[16] is the benchmark of the current federal learning framework. There are four steps in each round of FedAvg. First, the central server sends the global model to edge nodes. Second, the edge nodes update the model by its own data. The updated model can be used as a private model to serve the personal tasks of the nodes. Third, the edge nodes will upload the local model to the central server through encryption methods such as homomorphic encryption (HE)[17, 18] and differential privacy (DP)[19–21]. Last, the central server averages the weights of models uploaded by edge nodes as the new global model for the next round training.

Although federated learning effectively solves the data privacy issues of edge nodes during training, it still faces the following challenges: data, communications and security[22–25]. Our paper is mainly focuses on the offset of the federated learning model caused by noisy labels. Interestingly, the non-independent and identically distributed (IID) and unbalanced data distribution of the edge node has a similar impact on the global model as the label noise. Therefore, a brief review of non-IID and imbalance studies may help us to understand the federated noise learning more deeply.

The studies to solve data-ill issues in federated learning can be divided into two categories: improvement on local training or aggregation. Improvements in the aggregation process generally sacrifice communication efficiency or confidentiality, so we mainly focus on how to solve the data-ill problem in local training. Smith et al.[26] proposed federated multi-task learning, which can improve the generalization performance of the original task by sharing the representation of related tasks. FedProx[27] directly limits the local updates by ℓ_2-norm distance, while SCAFFOLD[28] corrects the local updates via variance reduction. Moon[29] utilizes the similarity between model representations to correct the local updating of edge nodes. Shen et al.[30] propose to use mutual learning to build a private model locally. Zhao et al.[31] propose a novel aggregation scheme that defends against Byzantine attacks by inverting local model updates.

There are a few studies for noise-tolerant FL[9–11, 32–34]. Both FOCUS[9] and Yang et al.[10] claim to be the first study for noise-tolerant federated learning. FOCUS[9] calculates the noise level of the edge node, thereby assigning the weight of each edge node when the model is aggregated. However, it requires a reliable and accurate dataset in the central server, and requires each node to upload cross-entropy information. Yang et al.[10] propose to interchange class-wise centroids of local data on each node to prevent local model from being corrupted by representations of noisy data. Tam et al.[11] mainly study the situation when some edge nodes contain noisy data and other nodes are completely trustworthy. Similar to FOCUS, it requires edge nodes to provide additional information to help model aggregation, which destroys the structure of federated learning so that it cannot be combined with existing FL encryption methods. Tour et al.[32] require a reliable and accurate dataset in the central server to refine the dataset of the edge node. Duan et al.[33] transform private data into privacy-preserving data representations on each client and identify clean data based on centralized data representations on the server. Fang and Ye[34] propose a robust aggregation scheme that can handle both noisy and heterogeneous clients.

2.2 Noise-tolerant learning

Classical deep learning implicitly assumes that the training data are sampled from a clean distribution, which is too restrictive for real-world scenarios. Therefore, label-noise learning has become very popular for both academia and industry[8, 14, 35, 36]. These studies can be summarized in the following three perspectives, including the data, the objective function and the optimization policy[5].

For data, the key is to construct the noise transition matrix[37–39] to link the noisy labels to clean labels. Sukhbaatar et al.[40] propose to use a constrained linear layer which is between the base network and cross-entropy loss layer to simulate the noise transition matrix. Following the linear case, Goldberger and Ben-Reuven[41] propose to use a non-linear network which can free of strong assumptions. Patrini et al.[42] introduce forward correction and backward correction to correct the outputs and loss. Based on forward correction, Hendrycks et al.[43] deal with severe noise situations by assuming that there is a small but trustworthy clean dataset.

For objective function, the key is to construct the noise-tolerant loss function which will reduce the network′s ability to fit complex labels while ensuring that the network will not underfit[35]. Namely, robust loss function make the network have the appropriate learning ability, in the case of learning clean labels, can not learn noisy labels due to its complexity. Zhang and Sabuncu[44] propose the generalized cross-entropy (GCE) loss function which combines the cross-entropy (CE) loss with the mean square error (MSE). Menon et al.[45] leverage gradient clipping to prevent over-confident. Wang et al.[46] propose symmetric cross entropy (SCE) inspired by the symmetric KL-divergence.

For optimization policy, the key is the use of memor-

ization effects for deep neural network. Arpit et al.[47] first reveale the memorization effect in neural networks, namely, the network always fits easy (clean) patterns first, and then as the number of iterations increases, it slowly fits complex (noisy) patterns. This phenomenon is also called small-loss trick[48]. MentorNet[48] is the first work to introduce the "small loss trick" into noise-tolerant learning. It utilizes a pre-trained network as a mentor to filter out noisy samples. However, the use of a fixed mentor leads to the accumulation of errors, ultimately hindering the identification of clean samples. Based on MentorNet, Han et al.[14] propose the co-teaching framework inspired by co-training. It randomly initializes two networks with the same structure but different parameters, and sends samples judged as clean sample by the current network to each other for learning and updating. Following co-teaching, Yu et al.[49] propose co-teaching+ to use the disagreement[50] with peer network keep two networks diverge.

3 Method

In this section, we introduce in detail the proposed method on noise-tolerant federated learning. For clarity, we only present the case where each edge node sends local model to the server in plaintext. Nevertheless, our method does not change the communication process in federated learning. Only the model updates are uploaded for each communication. Therefore, it can be easily combined with the existing federated learning encryption technology, such as homomorphic encryption (HE)[17, 18], differential privacy (DP)[19–21] and other technologies[51, 52]. We first provide the problem formulation, then elaborate the motivation and methodology of our proposed scheme. Finally, we offer the convergence analysis about the proposed federated Co-training.

3.1 Problem formulation

In this paper, we consider the k-class classification as the target task. Assume that $\mathcal{X} \subset \mathbf{R}^d$ is the feature space from which the examples are drawn, and $\mathcal{Y} = [k] = \{1, \cdots, k\}$ is the class label space. In the centralized classifier learning, we are given a training set $\mathcal{S} = \{(\boldsymbol{x}_1, y_1), \cdots, (\boldsymbol{x}_N, y_N)\}$, where $(\boldsymbol{x}_i, y_i)$ is drawn i.i.d. from an underlying distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$. The classifier is defined as a mapping function from feature space to label space $h(\boldsymbol{x}) = \arg\max_i f(\boldsymbol{x}; \Theta)_i$, where $f : \mathcal{X} \to \mathcal{C}$, $\mathcal{C} \subseteq [0, 1]^k$, $\forall \boldsymbol{c} \in \mathcal{C}$, $\mathbf{1}^{\mathrm{T}}\boldsymbol{c} = 1$. $f(\boldsymbol{x})$ denotes an approximation of $p(\cdot|\boldsymbol{x})$, which can be modeled by a neural network with parameters Θ ending with a softmax layer. The loss function to derive the optimal network parameters is defined as a mapping $L : \mathcal{C} \times \mathcal{Y} \to \mathbf{R}^+$.

When there exits noisy labels in the training set, the noise corruption process can be described as that a clean label y is flipped into a noisy version $\tilde{y}$ with probability $\eta_{\boldsymbol{x},\tilde{y}} = p(\tilde{y}|y, \boldsymbol{x})$[53]. The noisy L-risk is

$$R_L^{\eta}(f) = \mathbf{E}_{\mathcal{D}}[(1 - \eta_{\boldsymbol{x}})L(f(\boldsymbol{x}; \Theta), y) + \sum_{j \neq y} \eta_{\boldsymbol{x},j} L(f(\boldsymbol{x}; \Theta), j)] \tag{1}$$

where $\eta_{\boldsymbol{x}} = \sum_{j \neq y} \eta_{\boldsymbol{x},j}$ denotes the noise rate.

In the context of FL, the training set $\mathcal{S}$ is distributed in M local clients, i.e., $\mathcal{S} = \{\mathcal{S}_1, \cdots, \mathcal{S}_M\}$, and the corresponding noisy version is $\tilde{\mathcal{S}} = \{\tilde{\mathcal{S}}_1, \cdots, \tilde{\mathcal{S}}_M\}$. In different clients, the sample numbers $\{N_i\}_{i=1}^M$ and noise rates $\{\eta_i\}_{i=1}^M$ can be various, and the data distribution is usually non-i.i.d. Due to the privacy protection mechanism of FL, each client can only get access to data $\tilde{\mathcal{S}}_i = \{(\boldsymbol{x}_1, \tilde{y}_1), \cdots, (\boldsymbol{x}_{N_i}, \tilde{y}_{N_i})\}$ stored locally. The corresponding local noisy L-risk is defined as follows:

$$R_i(f) = \mathbf{E}_{\mathcal{D}_i}[(1 - \eta_{\boldsymbol{x}})L(f(\boldsymbol{x}; \Theta_{i,l}), y) + \sum_{j \neq y} \eta_{\boldsymbol{x},j} L(f(\boldsymbol{x}; \Theta_{i,l}), j)] \tag{2}$$

where $\mathcal{D}_i$ represents the data distribution of client i, and $\Theta_{i,l}$ denotes the local model parameters of client i. By minimizing $R_i(f)$, we can derive the specific local model with parameters $\Theta_{i,l}$ in the i-th client.

In the t-th training round of FL, all participating clients upload their model parameters $\{\Theta_{i,l}^{(t)}\}$ to the central server. The server then derives the parameters of the global model by aggregates all local ones by federated averaging:

$$\Theta^{(t+1)} = \frac{1}{M} \sum_{i=1}^{M} \Theta_{i,l}^{(t)} \tag{3}$$

from which it can be found that the robustness of the global model is directly affected by that of local models.

3.2 Noise-tolerant federated learning

The existing methods attempt to improve the noise robustness of the global model from the central server side by the aid of additional overhead information transmitted from the clients. These approaches, however, increase the communication burden as well as the risk of privacy leak. Instead, we turn to remedy the issue of noisy labels from the client side. Our scheme does not require any additional overhead information uploaded, and thus enjoys free lunch in communication.

3.2.1 Motivation

In the architecture of FL, the local model $f(\Theta_{i,l})$ is obtained by learning from local data, while the global model $f(\Theta)$ is obtained by aggregating from all local models. The difference in generation mechanism guarantees that there is inherent discrepancy of learning ability of the local and global models. Accordingly, they can serve as two very different classifiers. To integrate the strengths of these two classifiers, we exploit the well established co-training framework[13, 54], in which the two classifiers iteratively "teach each other" by exchanging

samples with their high confident predictions, so as to suppress the influence of noisy labels in model training. As theoretically proven in [15], the sufficient and necessary condition for co-training to be successful is as long as there is large diversity of two classifiers. The proposed federated co-training thus is well founded on the theoretical guarantee.

It is worth noting that, different from the counterpart co-teaching in centralized robust training[14] where two different networks should be maintained, in FL, the two classifiers are already there. We argue that co-training is more suitable for noise-tolerant federated learning.

3.2.2 Methodology

In the following, we elaborate the procedures of the proposed federated co-training. The framework is illustrated in Fig. 2.

In existing FL pipeline, for the t-th local training round, the i-th client downloads the global model $f(\Theta^{(t-1)})$ from the central server, which is used as the start point to train the local model. The approach however neglects the trained local model $f(\Theta_{i,l}^{(t-1)})$ in the last iteration. $f(\Theta^{(t-1)})$ represents the knowledge coming from other clients while $f(\Theta_{i,l}^{(t-1)})$ denotes the past experience of the local client i, which can be regarded as two complementary views. We propose to integrate them together through the following co-training paradigm:

Sample selection. As the two classifiers are with distinct network parameters, in each mini-batch, they will yield different predictions for the inputs $\{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_{N_i'}\}$, where N_i' denotes the number of samples in each mini-batch. We select samples that each classifier is confident about, i.e., more likely to be with correct labels, which are denoted as $\overline{\mathcal{S}}_{i,g}$ and $\overline{\mathcal{S}}_{i,l}$ respectively. The number of instances is controlled by $R(t)$, where t denotes the number of current communication round. The selection metric can be small loss as done in [14].

Model co-training. We then co-train the two classifiers by exchanging the selected samples, i.e., we update the global model using $\overline{\mathcal{S}}_{i,l}$ and update the local model using $\overline{\mathcal{S}}_{i,g}$. We learn both models by stochastic gradient descent (SGD):

$$\begin{aligned}\Theta_{i,g} &\leftarrow \Theta_{i,g} - \eta \nabla L\left(\overline{\mathcal{S}}_{i,l}; \Theta_{i,g}\right)\\ \Theta_{i,l} &\leftarrow \Theta_{i,l} - \eta \nabla L\left(\overline{\mathcal{S}}_{i,g}; \Theta_{i,l}\right)\end{aligned} \tag{4}$$

where $\Theta_{i,g}$ denotes the global model received by client i. Note that the global model and the local model are independently initialized at the beginning of each round, which ensures their divergency.

The above procedures are iteratively performed in each mini-batch to get the useful knowledge from each other. In summary, the workflow is shown in Algorithm 1.

Algorithm 1. Overhead-free noise tolerant FL

Input: The number of communication rounds T, the number of edge nodes M, learning rate η, the number of local epoch E, fixed τ

Output: The final global model Θ^T, the private model Θ_l^T in each edge node

Server executes:

Initialize Θ^0 and Θ_l^0 independently

for each round $t = 0, 1, \cdots, T$ **do**

  for $i = 1, 2, \cdots, M$ **in parallel do**

    Send the global model Θ^t to edge

    Node $\Theta_{i,g}^t \leftarrow \text{LocalUpdate}(i, \Theta^t)$

  $\Theta^{(t+1)} = \frac{1}{M}\sum_{i=1}^{M} \Theta_{i,g}^{(t)}$

Return Θ^T

LocalUpdate(i, Θ^t)

$\Theta_{i,g}^t \leftarrow \Theta^t$

for local epoch $j = 1, 2, \cdots, E$ **do**

  shuffle local training set S_i

  For each batch **do**

    Fetch mini-batch $\overline{S}_i$ from S_i

    $\overline{\mathcal{S}}_{i,g} \leftarrow \arg\min_{\mathcal{S}_i': |\mathcal{S}_i'| \geq R(t)|\overline{\mathcal{S}}_i|} \ell\left(g, \mathcal{S}_i'\right)$

    $\overline{\mathcal{S}}_{i,l} \leftarrow \arg\min_{\mathcal{S}_i': |\mathcal{S}_i'| \geq R(t)|\overline{\mathcal{S}}_i|} \ell\left(l, \mathcal{S}_i'\right)$

    $\Theta_{i,g} \leftarrow \Theta_{i,g} - \eta \nabla L\left(\overline{\mathcal{S}}_{i,l}; \Theta_{i,g}\right)$

    $\Theta_{i,l} \leftarrow \Theta_{i,l} - \eta \nabla L\left(\overline{\mathcal{S}}_{i,g}; \Theta_{i,l}\right)$

  $R(t) = 1 - \min\left\{\frac{t}{T}\tau, \tau\right\}$

return $\Theta_{i,g}^t$

3.2.3 Discussion

One critical issue in our method is how to ensure the divergency between the two models. The proposed method guarantees divergency from the following two aspects. First, it is worth noting that, in the context of federated learning, each client can only access its own local data and cannot share or exchange data with other clients. This means that each client learns a model based on its

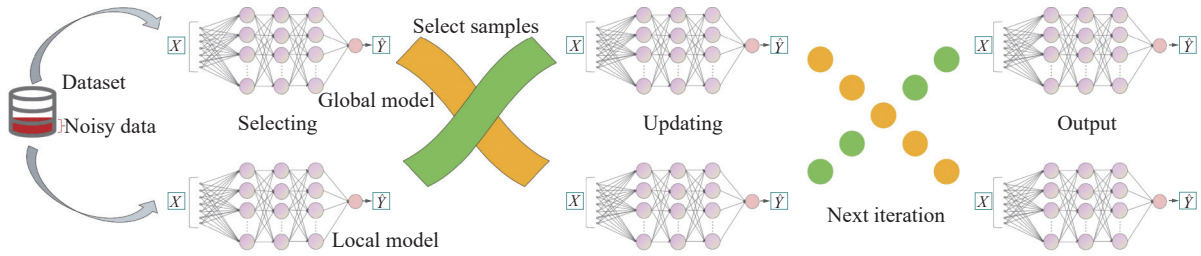


Fig. 2 Framework of our proposed noise-tolerant federated learning performed in each client. It follows the co-training paradigm, where the local model and the global model that own diverse learning ability serve as the two classifiers.

unique data view, rather than a global data view. Therefore, from this perspective, the global model and the local model can be considered as learned from two different views. Second, in the initial stage, the global model and the local model are independently initialized, so they have different initial parameters.

3.3 Convergence analysis

One negative effect of the noisy labels to FL is that they produce inconsistent decision boundaries among local clients, leading to individual local models difficult to reach consensus in the aggregation process of the global model. It would cost more communication rounds between central server and local clients. In this subsection, we provide convergence analysis about the traditional FL and the proposed noise-tolerant FL. As shown in Fig. 3, compared with the traditional FL, our proposed noise-tolerant FL can not only improve the classification accuracy, but also achieve faster convergence. It costs much fewer iterations to arrive at the same accuracy. For instance, when noise rate is 0.4, the traditional FedAvg costs about 50 rounds to achieve test accuracy 60%, while our method only costs about 10 rounds. This demonstrates that our scheme serves as a communication-friendly FL framework, which does not need to transmit any overhead message to from local to central, meanwhile reduces the communication cost between local and central.

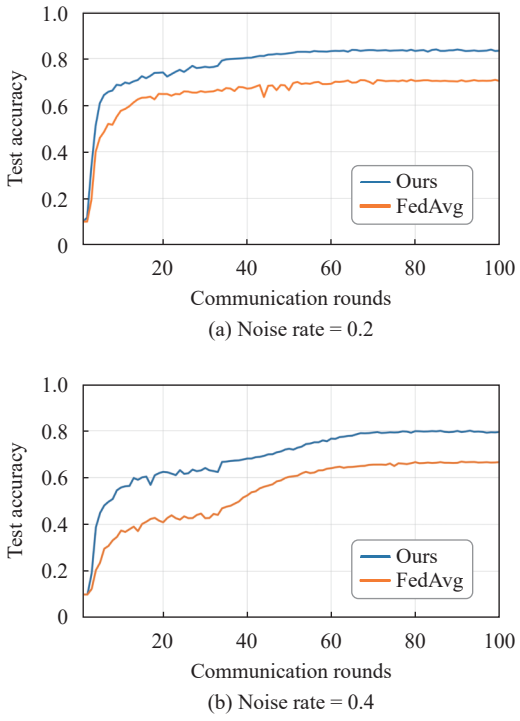


Fig. 3 Convergence comparison between traditional FL and our noise-tolerant FL. We investigate two levels of symmetric noise.

4 Experiments

In this section, extensive experimental results are provided to verify the effectiveness of the proposed algorithm.

4.1 Experiments setup

Datasets and label noise model. We verify the effectiveness of our approach on three benchmark datasets, MNIST, CIFAR10, CIFAR100, and one real-world dataset, Clothing1M[55], which are widely used for the evaluation of robustness to noisy labels and federated learning in previous studies[10, 35]. The details of these datasets are provided in Table 1. In our simulation, the training samples are randomly distributed on each client, and the central server only keep the test set to evaluate the final performance.

Table 1 Summary of data sets used in the experiments

	# of training	# of testing	# of class	Image size
MNIST	60 000	10 000	10	28×28
CIFAR-10	50 000	10 000	10	32×32
CIFAR-100	50 000	10 000	100	32×32
Clothing1M	927 591	72 409	14	227×227

We consider two types of label noise in our experiments: symmetric noise and asymmetric noise. The label noise can be modeled as

$$\tilde{y}_n = \begin{cases} i,\ i \in [k],\ i \neq y_n, & \text{with probability } \eta_{\boldsymbol{x}_n,i} \\ y_n, & \text{with probability } (1-\eta_{\boldsymbol{x}_n}) \end{cases} \tag{5}$$

where $\tilde{y}_n$ denotes the noisy label of $\boldsymbol{x}_n$, y_n denotes the corresponding true label. The probability of a label to be mislabeled is defined as $\eta_{\boldsymbol{x}_n}$.

4.2 Implementation details

We used the Pytorch framework to implement PyTorch[56], and conduct all the experiments on a NVIDIA 3090TI GPU. In our federated setting, We train the network with 100 global communication rounds and the epoch is set to 5. In our experiments, we implement the proposed method with three versions, which are conducted with different loss functions. Specifically, Ours-CE refers to the one using the traditional cross-entropy loss, which is known to be sensitive to label noise. Ours-GCE refers to the one using robust loss function GCE[44], which is used to demonstrate that our method can be easily combined with the well-developed robust loss function[8, 57, 58] to further improve the performance. The GCE loss defined as

$$\mathcal{L}_{GCE}\left(f(\boldsymbol{x}), \boldsymbol{e}_j\right) = \frac{\left(1 - f_j(\boldsymbol{x})^q\right)}{q} \tag{6}$$

where $q \in (0, 1]$. Ours-Sparse refers to the one using the most advanced method using sparse regularization[35], in which the loss function is defined as

$$\mathcal{L}_{Sparse}\left(f(\boldsymbol{x}), \boldsymbol{e}_j\right) = \mathcal{L}_{GCE}\left(f(\boldsymbol{x}), \boldsymbol{e}_j\right) + \lambda \left\| f\left(\boldsymbol{x}_i\right)\right\|_p^p. \tag{7}$$

4.3 Comparison study

We compared our proposed method with the following algorithms:

FedAvg[1], which is the classical architecture of federated learning that does not consider the influence of noisy labels to the learning processing. Since the code of this work is not available officially, we reproduced the code for comparison.

FedCo[14], which applies co-training directly to federated learning by using same initialized global and local models to identify noisy labels. We include FedCo in our comparison to demonstrate the superiority of using the naturally existing global model as the "classmate" model.

FedDR[33], which identifies clean data in federated learning by leveraging the correlation of global data representations. The approach involves transforming private data into privacy-preserving data representations on each client and then identifying clean data based on centralized data representations on the server. As the official code for this work is not publicly available, we reproduced the code for comparison.

Yang[10], which is a state-of-the art method that jointly consider federated learning and robust-tolerant learning. Since the code of this work is not available officially, we reproduced the code for comparison.

4.4 Evaluation on symmetric noise

Symmetric noise means that the mislabeling probability of all categories is equal, i.e., $\eta_{\boldsymbol{x}_n,i} = p,\ \forall x_n,\ i$. In the following, we provide the evaluation results on symmetric label noise.

Results on MNIST. Table 2 presents the accuracy on the MNIST testing set, which shows that all methods perform well when the noise rate is at the easiest level of $\epsilon = 0.2$. Even the standard federated learning method achieves a test accuracy of 89.12%, indicating the effectiveness of federated learning in dealing with noise. However, when the noise rate is increased to more challenging levels of $\epsilon = 0.4$ and $\epsilon = 0.6$, our proposed method outperforms all the other compared methods. In comparison with FedAvg, our method significantly improves accuracy and demonstrates its power in handling noisy labels in federated learning. Even the simplest version of our method, the Ours-CE, outperforms Yang′s method, which highlights the effectiveness of the co-training paradigm in the federated learning setting. Notably, FedDR[33] achieves an accuracy of 96.31%, which is higher than the Ours-CE version of our method that uses the CE loss, indicating the effectiveness of the FedDR approach in handling noisy labels. Overall, the experimental results demonstrate the superior performance of our proposed method in comparison to the state-of-the-art approaches in handling label noise in federated learning.

Results on CIFAR10/100. Table 3 presents the experimental results on the CIFAR10/100 testing sets,

Table 2 Test accuracy on the MNIST dataset with symmetric noise. We report the average accuracy over the last 5 rounds and the top 3 best results are **boldfaced**.

Noise ratio ϵ	FedAvg[1]	FedCo[14]	FedDR[33]	Yang[10]	Ours-CE	Ours-GCE	Ours-sparse
0.2	89.12	94.33	**96.31**	95.76	96.12	**96.56**	**97.21**
0.4	74.53	72.94	77.93	75.67	**79.89**	**81.21**	**83.87**
0.6	57.34	71.08	73.89	73.49	**75.23**	**77.86**	**78.27**

Table 3 Test accuracies (%) on the CIFAR10/100 dataset with symmetric noise. We report the average accuracy over the last 5 rounds and the top 3 best results are **boldfaced**.

Methods		Test accuracy (%)						
Datasets	ϵ	FedAvg[1]	FedCo[14]	FedDR[33]	Yang[10]	Ours-CE	Ours-GCE	Ours-sparse
CIFAR-10	0.2	70.49	76.14	80.33	80.62	**83.40**	**84.68**	**85.21**
	0.4	66.54	74.26	76.36	77.82	**79.43**	**81.65**	**82.47**
	0.6	49.23	71.37	73.21	72.68	**73.59**	**74.16**	**74.89**
CIFAR-100	0.2	46.64	51.44	54.26	55.23	**56.82**	**58.91**	**60.14**
	0.4	36.17	40.71	41.33	41.94	**44.73**	**47.97**	**50.59**
	0.6	23.72	33.49	35.17	35.79	**37.28**	**39.16**	**40.72**

which are more challenging than MNIST. For CIFAR100, our proposed method consistently outperforms all the other compared methods for different noise rates. Our method significantly improves accuracy compared to the baseline method, FedAvg. Notably, when the noise rate is 0.6, the simplest version of our method, Ours-CE, achieves a remarkable improvement of 13.46% in accuracy compared to FedAvg. Moreover, in comparison to FedDR[33], our Ours-CE version can improve the accuracy by 2.11%. These results demonstrate the superiority of our proposed method in handling label noise in federated learning, especially in challenging scenarios such as those present in CIFAR100. The findings underscore the potential of our approach in real-world applications, where label noise is a common issue.

As the experimental results in Tables 2 and 3 demonstrate, our proposed method takes advantage of the naturally different global and local networks in federated learning to perform co-training. By utilizing these two classifiers to select clean samples, our approach significantly outperforms FedDR[33] and Yang's method[10]. The comparison among our method with CE, GCE, and sparse loss clearly demonstrates that more powerful losses lead to even better performance. These findings underscore the effectiveness and versatility of our proposed approach in addressing label noise in federated learning. In real-world scenarios, where label noise is common, our method can serve as a promising solution for achieving high-performance federated learning.

4.5 Evaluation on asymmetric noise

Asymmetric noise means that the probabilities of mislabeling of all categories are not equal, i.e., $\exists i, j \in K$, $i \neq j$, $\eta_{\boldsymbol{x_n}, i} \neq \eta_{\boldsymbol{x_n}, j}$. The addition of asymmetric noise makes training a deep neural network more difficult. As shown in Table 4, the performance of standard federated learning is severely degraded in such scenarios. For instance, when the asymmetric noise rate increases to 0.4, the accuracy on CIFAR drops to below 40%. Although FedDR[33] and Yang's method can both improve performance compared to FedAvg, our method achieves the best performance among all compared methods. This can be attributed to the natural inconsistency between the global model and the local model in federated learning, which our method exploits to achieve better performance than the FedDR[33]. By leveraging the different characteristics of the global and local models, our method can better handle the challenge of asymmetric noise in federated learning.

4.6 Evaluation on non-IID distribution

In federated learning, non-IID distribution is also a common challenge, so we conduct experiments with noisy labels under non-IID scenarios. We use the Dirichlet distribution to create a non-IID data partition among clients. In particular, we sample $p_k \sim \mathrm{Dir}_N(\beta)$ and assign a fraction $p_{k,j}$ of the samples of class k to party j, where $\mathrm{Dir}(\beta)$ is the Dirichlet distribution with a concentration parameter β (0.5 by default). This partitioning strategy allows each client to have different (even zero) data samples in some classes. The experimental results are shown in Table 5. From the table, we can see that our method achieves the highest accuracy. When $\beta = 0.1$, our method's accuracy is 61.92%, which surpasses the best method by 2.35%. When $\beta = 5$, our method's accuracy is 64.26%, which surpasses the best method by 1.14%. The experimental results demonstrate that our method can handle both noisy labels and non-IID distribution simultaneously.

4.7 Evaluation on real-world noisy dataset

Furthermore, we evaluated our method on the widely used large-scale dataset Clothing1M[55], which consists of 14 clothing categories and contains 1 million images with noisy labels collected from several online shopping websites. In contrast to the artificially generated noise in CIFAR10/100, Clothing1M is a real-world dataset with a high degree of unknown structural noise, where the noise labels are dependent on both data features and class labels. Conducting experiments on real-world noisy datasets enables us to more objectively evaluate the perform-

Table 4 Test accuracies (%) of different methods on the CIFAR10/100 dataset with Asymmetric noise ($\epsilon \in \{0.2, 0.3, 0.4\}$). We report the average accuracy over the last 5 rounds and the top 3 best results are **boldfaced**.

Methods		Test accuracy (%)						
Datasets	ϵ	FedAvg[1]	FedCo[14]	FedDR[33]	Yang[10]	Ours-CE	Ours-GCE	Ours-sparse
CIFAR-10	0.2	73.67	74.53	78.22	78.26	**81.11**	**83.27**	**85.13**
	0.3	66.18	69.71	71.39	73.94	**75.09**	**78.23**	**80.12**
	0.4	56.26	57.32	59.22	60.14	**64.2**	**67.38**	**69.25**
CIFAR-100	0.2	56.83	57.27	57.94	58.73	**59.76**	**61.25**	**62.36**
	0.3	47.29	48.92	50.41	50.14	**52.36**	**54.84**	**56.93**
	0.4	38.36	39.47	39.79	41.29	**44.05**	**45.93**	**47.28**

Table 5 Test accuracies (%) of different methods on the CIFAR10 dataset under non-IID data distribution, and the noise rate is set to 0.3

Methods	Test accuracy (%)		
	$\beta = 0.1$	$\beta = 0.5$	$\beta = 5$
FedAvg	56.74	58.93	60.07
FedCo	57.68	60.17	61.44
FedDR	59.12	61.94	62.79
Yang	59.57	62.21	63.12
Ours-CE	**61.92**	**63.37**	**64.26**

Table 6 Test accuracies (%) of different methods on the Clothing1M dataset

Datasets	Method	Test accuracy (%)
Clothing1M	FedAvg[1]	71.63
	FedCo[14]	72.24
	FedDR[33]	72.33
	Yang[10]	74.64
	Ours-CE	**76.13**
	Ours-GCE	**77.25**
	Ours-Sparse	**77.94**

ance of the proposed algorithm. The results in Table 6 demonstrate that FedDR[33] slightly outperforms FedAvg on this challenging dataset. Nevertheless, our method still outperforms the state-of-the-art methods. Compared with FedDR[33] and Yang's method[10], our method achieves up to 5.61% and 3.3% gains, respectively. This indicates the superiority of our method in handling real-world noisy datasets.

4.8 Limitations

Our proposed method has the advantage of being communication-efficient. It is able to perform noise-tolerant federated learning without sending any additional information to the central server, unlike some existing approaches that require extra overhead information. However, this overhead-free property comes at a cost of increased computational burden on the local clients. This is because our method requires additional co-training to be performed on the local clients. Therefore, our scheme is more appropriate for cross-silo federated learning where the computation burden of local clients is not a bottleneck. However, it may not be as suitable for cross-device federated learning where the computational limitations of the individual devices need to be taken into account.

5 Conclusions

In this paper, we proposed an overhead-free noise-tolerant federated learning framework, which employs inherent discrepancy of learning ability of the local and global models in FL. Different from the previous research, our method can protect the data privacy of edge nodes while handling noisy data. Experiments have shown that the effect of using the global model and local model that naturally exist in federated learning as "classmates" is much better than randomly initializing two networks with different parameters. And we also show the inclusiveness of the proposed algorithm, that is, it can be easily combined with the most advanced robust loss function in the field of label-noise learning to achieve better performance. Additionally, our proposed method will provide a private local model for each edge node after the training is completed, which may help the non-IID issue. We will jointly consider the label-noise learning and non-IID in future research.

Acknowledgements

This work was supported by National Natural Science Foundation of China (Nos. 92270116 and 62071155).

Declarations of conflict of interest

The authors declared that they have no conflicts of interest to this work.

References

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, B. A. Y. Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, Fort Lauderdale, USA, vol. 54, pp. 1273–1282, 2017.

[2] T. Lin, L. J. Kong, S. U. Stich, M. Jaggi. Ensemble distillation for robust model fusion in federated learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, Vancouver, Canada, Article number 198, 2020.

[3] R. Shokri, V. Shmatikov. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, Denver, USA, pp. 1310–1321, 2015. DOI: 10.1145/2810103.2813687.

[4] C. Y. Zhang, S. Bengio, M. Hardt, B. Recht, O. Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, vol. 64, no. 3, pp. 107–115, 2021. DOI: 10.1145/3446776.

[5] B. Han, Q. M. Yao, T. L. Liu, G. Niu, I. W. Tsang, J. T. Kwok, M. Sugiyama. A survey of label-noise representation learning: Past, present and future, [Online], Available: https://arxiv.org/abs/2011.04406, 2020.

[6] H. Song, M. Kim, D. Park, Y. Shin, J. G. Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, to be published. DOI: 10.1109/TNNLS.2022.3152527.

[7] D. Karimi, H. R. Dou, S. K. Warfield, A. Gholipour. Deep learning with noisy labels: Exploring techniques and remedies in medical image analysis. *Medical Image Analysis*, vol. 65, Article number 101759, 2020. DOI: 10.1016/j.media.2020.101759.

[8] X. Zhou, X. M. Liu, J. J. Jiang, X. Gao, X. Y. Ji. Asymmetric loss functions for learning with noisy labels. In *Proceedings of the 38th International Conference on Machine Learning*, vol. 139, pp. 12846–12856, 2021.

[9] Y. Q. Chen, X. D. Yang, X. Qin, H. Yu, B. Chen, Z. Q. Shen. FOCUS: Dealing with label quality disparity in federated learning, [Online], Available: https://arxiv.org/abs/2001.11359, 2020.

[10] S. Yang, H. Park, J. Byun, C. Kim. Robust federated learning with noisy labels. *IEEE Intelligent Systems*, vol. 37, no. 2, pp. 35–43, 2022. DOI: 10.1109/MIS.2022.3151466.

[11] K. Tam, L. Li, B. Han, C. Z. Xu, H. Z. Fu. Federated noisy client learning, [Online], Available: https://arxiv.org/abs/2106.13239, 2021.

[12] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, R. G. L. D'Oliveira, H. Eichner, S. E. Rouayheb, D. Evans, J. Gardner, Z. Garrett, A. Gascón, B. Ghazi, P. B. Gibbons, M. Gruteser, Z. Harchaoui, C. Y. He, L. He, Z. Y. Huo, B. Hutchinson, J. Hsu, M. Jaggi, T. Javidi, G. Joshi, M. Khodak, J. Konecný, A. Korolova, F. Koushanfar, S. Koyejo, T. Lepoint, Y. Liu, P. Mittal, M. Mohri, R. Nock, A. Özgür, R. Pagh, H. Qi, D. Ramage, R. Raskar, M. Raykova, D. Song, W. K. Song, S. U. Stich, Z. T. Sun, A. T. Suresh, F. Tramèr, P. Vepakomma, J. Y. Wang, L. Xiong, Z. Xu, Q. Yang, F. X. Yu, H. Yu, S. Zhao. Advances and open problems in federated learning. *Foundations and Trends.® in Machine Learning*, vol. 14, pp. 1–210, 2021. DOI: 10.1561/2200000083.

[13] A. Blum, T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the 11th Annual Conference on Computational Learning Theory*, ACM, Madison, USA, pp. 92–100, 1998. DOI: 10.1145/279943.279962.

[14] B. Han, Q. M. Yao, X. R. Yu, G. Niu, M. Xu, W. H. Hu, I. W. Tang, M. Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, Montreal, Canada, pp. 8536–8546, 2018.

[15] W. Wang, Z. H. Zhou. A new analysis of co-training. In *Proceedings of the 27th International Conference on Machine Learning*, Omnipress, Israel, pp. 1135–1142, 2010.

[16] S. U. Stich. Local SGD converges fast and communicates little. In *Proceedings of the 7th International Conference on Learning Representations*, New Orleans, USA, pp. 1–19, 2019.

[17] C. L. Zhang, S. Y. Li, J. Z. Xia, W. Wang, F. Yan, Y. Liu. BatchCrypt: Efficient homomorphic encryption for cross-silo federated learning. In *Proceedings of USENIX Annual Technical Conference*, Article number 33, 2020.

[18] S. Hardy, W. Henecka, H. Ivey-Law, R. Nock, G. Patrini, G. Smith, B. Thorne. Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption, [Online], Available: https://arxiv.org/abs/1711.10677, 2017.

[19] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. S. Quek, H. V. Poor. Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3454–3469, 2020. DOI: 10.1109/tifs.2020.2988575.

[20] R. C. Geyer, T. Klein, M. Nabi. Differentially private federated learning: A client level perspective, [Online], Available: https://arxiv.org/abs/1712.07557, 2017.

[21] H. B. McMahan, D. Ramage, K. Talwar, L. Zhang. Learning differentially private recurrent language models. In *Proceedings of the 6th International Conference on Learning Representations*,Vancouver, Canada, 2018.

[22] F. L. Zhang, Y. C. Li, S. Y. Lin, Y. F. Shao, J. J. Jiang, X. M. Liu. Large sparse kernels for federated learning, [Online], Available: https://openreview.net/forum?id=ZCv4E1unfJP, 2023.

[23] Q. Yang, Y. Liu, T. J. Chen, Y. X. Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology*, vol. 10, no. 2, Article number 12, 2019. DOI: 10.1145/3298981.

[24] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. T. Jiao, Y. C. Liang, Q. Yang, D. Niyato, C. Y. Miao. Federated learning in mobile edge networks: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 2031–2063, 2020. DOI: 10.1109/COMST.2020.2986024.

[25] T. Li, A. K. Sahu, A. Talwalkar, V. Smith. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 50–60, 2020. DOI: 10.1109/MSP.2020.2975749.

[26] V. Smith, C. K. Chiang, M. Sanjabi, A. S. Talwalkar. Federated multi-task learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, USA, vol. 30, pp. 4427–4437, 2017

[27] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, V. Smith. Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems*, Austin, USA, pp. 429–450, 2020.

[28] S. P. Karimireddy, S. Kale, M. Mohri, S. J. Reddi, S. U. Stich, A. T. Suresh. SCAFFOLD: Stochastic controlled averaging for federated learning. In *Proceedings of the 37th International Conference on Machine Learning*, vol. 119, pp. 5132–5143, 2020.

[29] Q. B. Li, B. S. He, D. Song. Model-contrastive federated learning. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Nashville, USA, pp. 10708–10717, 2021. DOI: 10.1109/CVPR46437.2021.01057.

[30] T. Shen, J. Zhang, X. K. Jia, F. D. Zhang, G. Huang, P. Zhou, K. Kuang, F. Wu, C. Wu. Federated mutual learning, [Online], Available: https://arxiv.org/abs/2006.16765, 2020.

[31] B. Zhao, P. Sun, T. Wang, K. Y. Jiang. FedInv: Byzantine-robust federated learning by inversing local model updates. In *Proceedings of the 36th AAAI conference on Artificial Intelligence*, California, USA, pp. 9171–9179, 2022. DOI: 10.1609/aaai.v36i8.20903.

[32] T. Tuor, S. Q. Wang, B. J. Ko, C. C. Liu, K. K. Leung. Overcoming noisy and irrelevant data in federated learning. In *Proceedings of the 25th International Conference on Pattern Recognition*, IEEE, Milan, Italy, pp. 5020–5027, 2021. DOI: 10.1109/ICPR48806.2021.9412599.

[33] S. M. Duan, C. Y. Liu, Z. S. Cao, X. P. Jin, P. Y. Han. Fed-DR-Filter: Using global data representation to reduce the impact of noisy labels on the performance of federated learning. *Future Generation Computer Systems*, vol. 137, pp. 336–348, 2022. DOI: 10.1016/j.future.2022.07.013.

[34] X. W. Fang, M. Ye. Robust federated learning with noisy and heterogeneous clients. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, New Orleans, USA, pp. 10062–10071, 2022. DOI: 10.1109/CVPR52688.2022.00983.

[35] X. Zhou, X. M. Liu, C. Y. Wang, D. M. Zhai, J. J. Jiang, X. Y. Ji. Learning with noisy labels via sparse regularization. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, IEEE, Montreal, Canada, pp. 72–81, 2021. DOI: 10.1109/ICCV48922.2021.00014.

[36] X. Zhou, X. M. Liu, D. M. Zhai, J. J. Jiang, X. Y. Ji. Asymmetric loss functions for noise-tolerant learning: Theory and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 7, pp. 8094–8109, 2023. DOI: 10.1109/TPAMI.2023.3236459.

[37] A. K. Menon, B. van Rooyen, N. Natarajan. Learning from binary labels with instance-dependent corruption, [Online], Available: https://arxiv.org/abs/1605.00751, 2016.

[38] X. B. Xia, T. L. Liu, N. N. Wang, B. Han, C. Gong, G. Niu, M. Sugiyama. Are anchor points really indispensable in label-noise learning?. In *Proceedings of the 33rd International Conference o Neural Information Processing Systems*, Vancouver, Canada, Article number 614, 2019.

[39] J. C. Cheng, T. L. Liu, K. Ramamohanarao, D. C. Tao. Learning with bounded instance and label-dependent label noise. In *Proceedings of the 37th International Conference on Machine Learning*, Vienna, Austria, vol. 119, pp. 1789–1799, 2020.

[40] S. Sukhbaatar, J. Bruna, M. Paluri, L. Bourdev, R. Fergus. Training convolutional networks with noisy labels, [Online], Available: https://arxiv.org/abs/1406.2080, 2014.

[41] J. Goldberger, E. Ben-Reuven. Training deep neural-networks using a noise adaptation layer. In *Proceedings of the 5th International Conference on Learning Representations*, Toulon, France, 2017.

[42] G. Patrini, A. Rozza, A. K. Menon, R. Nock, L. Z. Qu. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, USA, pp. 2233–2241, 2017. DOI: 10.1109/CVPR.2017.240.

[43] D. Hendrycks, M. Mazeika, D. Wilson, K. Gimpel. Using trusted data to train deep networks on labels corrupted by severe noise. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, Montreal, Canada, pp. 10477–10486, 2018.

[44] Z. L. Zhang, M. R. Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, Montreal, Canada, pp. 8792–8802, 2018.

[45] A. K. Menon, A. S. Rawat, S. J. Reddi, S. Kumar. Can gradient clipping mitigate label noise? In *Proceedings of the 8th International Conference on Learning Representations*, Addis Ababa, Ethiopia, 2020.

[46] Y. S. Wang, X. J. Ma, Z. Y. Chen, Y. Luo, J. F. Yi, J. Bailey. Symmetric cross entropy for robust learning with noisy labels. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, IEEE, Seoul, Republic of Korea, pp. 322–330, 2019. DOI: 10.1109/ICCV.2019.00041.

[47] D. Arpit, S. Jastrzębski, N. Ballas, D. Krueger, E. Bengio, M. S. Kanwal, T. Maharaj, A. Fischer, A. Courville, Y. Bengio, S. Lacoste-Julien. A closer look at memorization in deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, Sydney, Australia, vol. 70, pp. 233–242, 2017.

[48] L. Jiang, Z. Y. Zhou, T. Leung, L. J. Li, L. Fei-Fei. MentorNet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *Proceedings of the 35th International Conference on Machine Learning*, Stockholm, Sweden, vol. 80, pp. 2309–2318, 2018.

[49] X. R. Yu, B. Han, J. C. Yao, G. Niu, I. W. Tsang, M. Sugiyama. How does disagreement help generalization against label corruption?. In *Proceedings of the 36th International Conference on Machine Learning*, Long Beach, USA, vol. 97, pp. 7164–7173, 2019.

[50] E. Malach, S. Shalev-Shwartz. Decoupling "when to update" from "how to update". In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, USA, pp. 961–971, 2017.

[51] F. Mo, A. S. Shamsabadi, K. Katevas, S. Demetriou, I. Leontiadis, A. Cavallaro, H. Haddadi. DarkneTZ: Towards model privacy at the edge using trusted execution environments. In *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*, ACM, Toronto, Canada, pp. 161–174, 2020. DOI: 10.1145/3386901.3388946.

[52] F. Mo, H. Haddadi, K. Katevas, E. Marin, D. Perino, N. Kourtellis. PPFL: Privacy-preserving federated learning with trusted execution environments. In *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services*, ACM, pp. 94–108, 2021. DOI: 10.1145/3458864.3466628.

[53] A. Ghosh, H. Kumar, P. S. Sastry. Robust loss functions under label noise for deep neural networks. In *Proceedings of the 31st AAAI conference on Artificial Intelligence*, San Francisco, USA, pp. 1919–1925, 2017. DOI: 10.1609/aaai.v31i1.10894.

[54] S. Y. Qiao, W. Shen, Z. S. Zhang, B. Wang, A. Yuille. Deep co-training for semi-supervised image recognition. In *Proceedings of the 15th European Conference on Computer Vision*, Springer, Munich, Germany, pp. 142–159, 2018. DOI: 10.1007/978-3-030-01267-0_9.

[55] T. Xiao, T. Xia, Y. Yang, C. Huang, X. G. Wang. Learning from massive noisy labeled data for image classification. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Boston, USA, pp. 2691–2699, 2015. DOI: 10.1109/CVPR.2015.7298885.

[56] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. M. Lin, N. Gimelshein, L. Antiga,

A. Desmaison, A. Köpf, E. Z. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. J. Bai, S. Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Vancouver, Canada, Article number 721, 2019.

[57] Y. Kim, J. Yim, J. Yun, J. Kim. NLNL: Negative learning for noisy labels. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, IEEE, Seoul, Republic of Korea, pp. 101–110, 2019. DOI: 10.1109/ICCV.2019.00019.

[58] X. J. Ma, H. X. Huang, Y. S. Wang, S. Romano, S. M. Erfani, J. Bailey. Normalized loss functions for deep learning with noisy labels. In *Proceedings of the 37th International Conference on Machine Learning*, Vienna, Austria, vol. 119, pp. 6543–6553, 2020.

Shiyi Lin received the B. Eng. and M. Eng. degrees in instrument science and technology from Harbin Institute of Technology, China in 2018 and 2020, respectively. She is currently a Ph. D. degree candidate in computer science at Harbin Institute of Technology (HIT), China.

Her research interests include unsupervised learning and federated learning.

E-mail: shiyi.lin.hit@outlook.com
ORCID iD: 0009-0003-2668-6688

Deming Zhai received the B. Sc., M. Sc. and Ph. D. (Hons.) degrees in computer science from Harbin Institute of Technology (HIT), China in 2007, 2009, and 2014, respectively. She is currently an associate professor with Department of Computer Science, HIT, China. In 2011, she was with Hong Kong University of Science and Technology, China, as a visiting student. In 2012, she was with the GRASP Laboratory, University of Pennsylvania, USA, as a visiting scholar. From August 2014 to April 2016, she worked as a project researcher at National Institute of Informatics (NII), Japan.

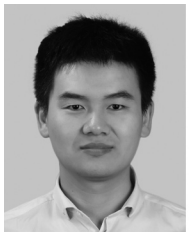
Her research interests include machine learning and its application in computer version.

E-mail: zhaideming@hit.edu.cn

Feilong Zhang received the B. Eng. and M. Eng. degrees in instrument science and technology from Harbin Institute of Technology, China in 2018 and 2020, respectively. He is currently a Ph. D. degree candidate in computer science from Harbin Institute of Technology (HIT), China.

His research interests include computer vision, machine learning and federated learning.

E-mail: flzhang.hit@gmail.com



Junjun Jiang received the B. Sc. degree in mathematics from Department of Mathematics, Huaqiao University, China in 2009, and the Ph. D. degree in computer science from School of Computer, Wuhan University, China in 2014. From 2015 to 2018, he was an associate professor at China University of Geosciences, China. Since 2016, he has been a project researcher with National Institute of Informatics, Japan. He is currently a professor with School of Computer Science and Technology, Harbin Institute of Technology, China. He won the Finalist of the World's FIRST 10K Best Paper Award at ICME 2017, and the Best Student Paper Runner-up Award at MMM 2015. He received the 2016 China Computer Federation (CCF) Outstanding Doctoral Dissertation Award and 2015 ACM Wuhan Doctoral Dissertation Award, China.

His research interests include image processing and computer vision.

E-mail: jiangjunjun@hit.edu.cn

Xianming Liu received the B. Sc., M. Sc. and Ph. D. degrees in computer science from Harbin Institute of Technology (HIT), China in 2006, 2008 and 2012, respectively. In 2011, he spent half a year at Department of Electrical and Computer Engineering, McMaster University, Canada, as a visiting student, where he was a post-doctoral fellow from 2012 to 2013. He was a project researcher with National Institute of Informatics (NII), Japan from 2014 to 2017. He is currently a professor with School of Computer Science and Technology, HIT, China. He has published over 50 international conference and journal publications, including top IEEE journals, such as T-IP, T-CSVT, T-IFS, and T-MM, and top conferences, such as ICML, ICLR, CVPR, ICCV, etc. He was a recipient of the IEEE ICME 2016 Best Student Paper Award.

His research interests include trustworthy AI, computational imaging, biomedical signal compression and 3D signal processing and analysis.

E-mail: csxm@hit.edu.cn (Corresponding author)
ORCID iD: 0000-0002-8857-1785

Xiangyang Ji received the B. Sc. degree in materials science and the M. Sc. degree in computer science from Harbin Institute of Technology, China in 1999 and 2001, respectively, and the Ph. D. degree in computer science from Institute of Computing Technology, Chinese Academy of Sciences, China in 2008. He joined Tsinghua University, China in 2008, where he is currently a professor with Department of Automation, School of Information Science and Technology. He has authored over 100 referred conference and journal papers.

His research interests include signal processing, image/video compressing, and intelligent imaging.

E-mail: xyji@tsinghua.edu.cn