ECAI 2023
K. Gal et al. (Eds.)

doi:10.3233/FAIA230529

FedCoop: Cooperative Federated Learning for Noisy Labels

Kahou Tam[a], Li Li[a;*], Yan Zhao[b] and Chengzhong Xu[a]

[a]State Key Lab of IoTSC, University of Macau.
[b]Bytedance Inc.

Abstract. Federated Learning coordinates multiple clients to collaboratively train a shared model while preserving data privacy. However, the training data with noisy labels located on the participating clients severely harm the model performance. In this paper, we propose FedCoop, a cooperative Federated Learning framework for noisy labels. FedCoop mainly contains three components and conducts robust training in two phases, data selection and model training. In the data selection phase, in order to mitigate the confirmation bias caused by a single client, the Loss Transformer intelligently estimates the probability of each sample's label to be clean through cooperating with the helper clients, which have high data trustability and similarity. After that, the Feature Comparator evaluates the label quality for each sample in terms of latent feature space in order to further improve the robustness of noisy label detection. In the model training phase, the Feature Matcher trains the model on both the noisy and clean data in a semi-supervised manner to fully utilize the training data and exploits the feature of global class to increase the consistency of pseudo labeling across the clients. The experimental results show FedCoop outperforms the baselines on various datasets with different noise settings. It effectively improves the model accuracy up to 62% and 27% on average compared with the baselines.

1 Introduction

Federated Learning (FL) is designed to coordinate multiple clients in order to train a shared model collaboratively while guaranteeing data privacy [16, 25]. It has great potential to well support a wide range of applications including medical and financial services. Despite the promising benefits, the following critical obstacle severely prevents FL from being effectively deployed in real-world scenarios. Due to the fact that high-quality annotation is extremely expensive and time-consuming [2], the data located on different clients usually contain noisy labels with different ratios. For instance, the ratio of noisy labels in real-world datasets is reported to range from 8.0% to 38.5% [31]. In FL, the local models can easily overfit on corrupted data and undermine the performance of the shared model via model aggregation [22].

Limitations of Prior Art. In order to mitigate the impact of noisy labels during the learning process, different approaches have been proposed in both centralized and federated learning scenarios. Prior solutions in centralized settings, such as designing robust loss functions [29, 12] and sample selection strategies to identify noise-free samples [13, 23, 31, 28], have shown promise for effective training with noisy labels. However, they are not directly transferable to FL for the two fundamental issues: 1) highly heterogeneous noise distribution among clients causes local model divergence that cannot be addressed by robust loss functions [22]; 2) sample selection approaches for centralized setting are infeasible when dealing with limited training data on each client, which overfit the noisy data and lead to unstable performance. Recently, several methods have been proposed to mitigate the problem of noisy labels in FL [26, 6, 35, 34, 33]. These methods mainly focus on obtaining data with clean labels by modeling noise probability or utilizing the memorization effect of Deep Neural Networks (DNNs). However, noise probability modeling methods require publicly available clean benchmark data to model the noise samples, which is often difficult to obtain in real-world scenarios [30, 31]. To overcome the restriction of public benchmark data, some methods employ the distinction of samples' loss through the memorization effect of DNNs to filter out the noisy samples. Nevertheless, these methods neglect the issue of confirmation bias and the overfitting property of DNNs, which severely impact the effectiveness of sample selection [35, 34]. Therefore, a new FL framework that can train a shared model effectively with clients having noisy labels, without public benchmark data is urgently required.

Challenges. Designing such a robust training framework is challenging for the following reasons in FL. Firstly, in contrast to the vast reservoirs of training data accessible for centralized learning, the quantity of client data in FL is typically limited. Thus, the client's local model can quickly overfit the samples with noisy labels when the amount of training data is limited [1]. As a result, the local model can memorize all the samples with noisy labels, leading to limited discriminatory capability of the empirical loss between noisy and clean labels. Thus, how to correctly distinguish the data with noisy labels using the memorization effect with limited amount of data becomes the first critical challenge. Secondly, to preserve privacy, the local data in FL cannot be accessed by the server. Thus, the client must select confident data by itself, which can lead to confirmation bias. This bias arises when the client unhesitatingly selects the data with noisy labels for training [24]. The confirmation bias can mislead the convergence direction of the local model. Therefore, how to conduct data selection to effectively avoid confirmation bias is the second critical challenge. Furthermore, data heterogeneity is a critical problem in FL, leading to severe model divergence across clients [16]. Learning only from data with clean labels will further exacerbate the divergence across clients and lose the valuable features of the data

* Corresponding Author. Email: llili@um.edu.mo

with noisy labels. Therefore, how to effectively utilize the heterogeneous data with noisy labels to train the robust and general global model is the third critical challenge.

In this paper, we propose FedCoop, a cooperative federated learning framework for noisy labels. FedCoop is a two-stage framework that mainly consists of three components: 1) Loss Transformer, 2) Feature Comparator, and 3) Feature Matcher. In each training round, each client first performs data selection, where the Loss Transformer and Feature Comparator jointly evaluate label quality of the samples within it. Specifically, in the Loss Transformer, an inter-client integrated loss is designed to well estimate the probability of each sample's label being clean with the help of other clients. The Feature Comparator evaluates the label quality for each sample in terms of latent feature space in order to further improve the robustness of noisy label detection. After that, the client's local dataset is dynamically divided into clean-labels and noisy-labels datasets by integrating the evaluation information from these two components. After data selection, to effectively learn from the data with noisy labels, we design the Feature Matcher, a feature-based Semi-Supervised Learner (SSL), which is proposed to exploit the feature of global class to increase the accuracy of pseudo labeling and reduce the model inconsistency across the clients.

Our main contributions are summarized as follows:

- We propose a general noisy label detection and robust global model training framework for Federated Learning, FedCoop, which integrates the robust and general noisy label detection method with an FL-adaptive semi-supervised learning scheme for federated noisy label learning.
- We fully exploit the cooperative advantage in FL to develop the Loss Transformer, which identifies the clients' noisy label data by inter-client integrated loss to prevent the clients from trapping in confirmation bias.
- We address the issue of DNNs rapidly overfitting on a small amount of noisy data by designing FilterNet as a DNN with limited capacity in the Loss Transformer.
- We design the Feature Comparator to improve the robustness of clean label selection, which identifies the noisy label by representation discrepancy across the clean label data and noisy label data.
- We propose the Feature Matcher, the feature-based SSL scheme to adapt the heterogeneous noise degree in FL by consistently pseudo-labeling guided by global feature. The experiment results show that FedCoop improves the model accuracy up to 62% and 27% on average compared with the baselines.

2 Related Work

Label-noise Learning in Centralized Training. Existing works in centralized training can be divided into two categories: 1) Noise cleaning-based approaches [13, 4, 18] and 2) Training noise-robust models [12, 29]. The noise cleaning-based methods first select out the clean data and then conduct the model training based on them [13, 23, 18]. For instance, Tanaka et al.[23] integrate label correction to relabel the noisy samples in order to improve the efficiency of data utilization. However, these sample selection approaches are not effective for FL since they assume that the data with clean labels have small losses. In FL, the limited amount of data available in each client makes it easier for the local model to overfit the data with noisy labels. As a result, data with noisy labels may have smaller losses, leading to potential issues with model generalization. The other category of methods [12, 29] mainly focus on designing robust loss functions to train robust models using the data with noisy

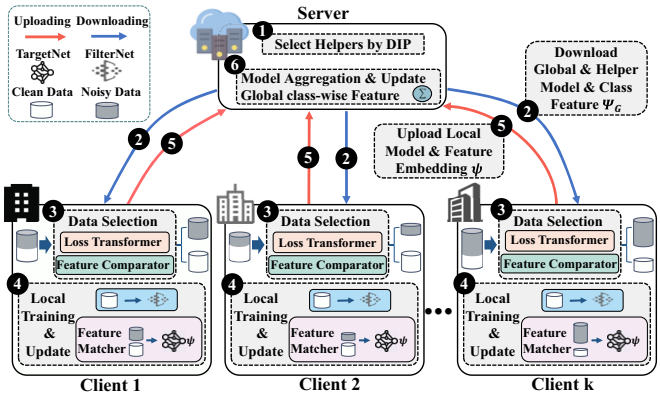


Figure 1. An overview of FedCoop.

labels. The performance of this type of methods decreases seriously as the ratio of noise increases. As the noise distributions among the clients are usually different, these approaches lead to severe weight divergence across the local models.

Federated Learning with Noisy Label. Existing works about FL with noisy labels can be summarized into two main categories: 1) Benchmark data-dependent methods [9, 26, 6] and 2) Label correction-based methods [33, 34, 35]. The Benchmark data-dependent methods extract the subset of clean clients or data with clean labels using public benchmark data. For instance, DS [26] selects the confident samples by estimating the similarity between the client's training data and benchmark data from the server. However, these approaches highly rely on the benchmark data, which is hard to retrieve due to the privacy issue. The label correction-based methods first train the global model with the client's confident samples and then performs the label correction on the client's data with noisy labels. For instance, RoFL [34] utilizes label correction while naively training the sample with small loss to create local centroids and exchange them between clients and servers. However, these methods have the following critical limitations. First, they don't consider the overfitting property of DNNs and confirmation bias problem [28] when designing their sample selection approaches, which severely impacts the effectiveness of sample selection. Second, they are not designed for heterogeneous noise distributions across different clients. Moreover, the methods mentioned above do not fully exploit the data with noisy labels to train the global model, resulting in poor generalization of the global model.

3 Method

3.1 Preliminaries and Definitions

We consider a cross-silo FL system with K participating clients and one global server. Unlike the cross-device FL, the clients of cross-silo FL usually have sufficient computational resources and stable network connections [19]. Let θ^G be the parameters of a global model, and $L = \{\theta^k\}_{k=1}^K$ be the set of local models for K participants. Let $p_\theta(x)$ be the class probabilities predicted by the model θ given the input x. We denote $\mathrm{CE}(p, q)$ as the cross-entropy between two distributions p and q. The training dataset of client k is denoted by $D^k = \{(x_i^k, y_i^k)\}_{i=1}^{N_k}$, where N_k is the total number of training samples for the client. As we investigate the noisy label issue in FL, the label y_i^k of each sample in client k can be noisy $\tilde{y}_i^k$, but the feature of training instance x_i^k is clean. We denote the set of noise ratios for K clients as $\{\epsilon_k\}_{k=1}^K$.

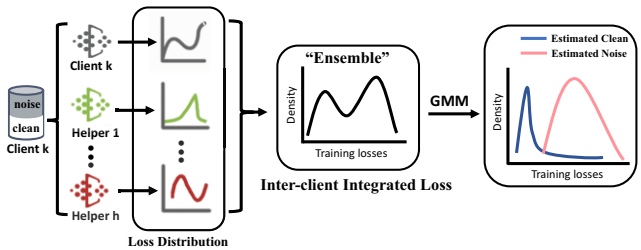


Figure 2. The workflow of Loss Transformer.

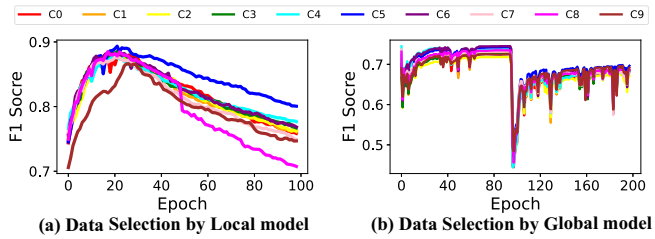


Figure 3. The F1 score of the centralized noisy data detection methodology on CIFAR-10 under 50% symmetric noise in FL. Centralized data selection methodology is performed for 10 clients by (a) the local model; (b) the global model.

3.2 *Overview*

Figure 1 shows the overview of FedCoop. Following the standard schema of FL, we design three novel components to achieve robust learning: **Loss Transformer** (Section 3.3), **Feature Comparator** (Section 3.4), and **Feature Matcher** (Section 3.6). These components are deployed on the client's side and intelligently interact with the central server in order to conduct robust training in two phases: data selection and model training. In the data selection phase, during each training round, the server first selects the helper clients for the client's Loss Transformer using noisy model discrepancy (DIP), which jointly measures the noise level and model similarity between target client and candidate clients. Then, the server sends the global model and the global class-wise features to the clients. Before local training starts, each client performs data selection by Loss Transformer and Feature Comparator. The Loss Transformer utilizes the FilterNet, a DNN with limited capacity, to evaluate the label quality while cooperating with the helper clients to mitigate the confirmation bias by the client itself. The Feature Comparator evaluates the label quality through comparing each sample's latent feature extracted by TargetNet (the model requiring training through FL) with the global class-wise features. After splitting the dataset into noisy and clean data, FilterNet and TargetNet are simultaneously trained by the separated data to prevent error accumulation and make the system more robust to noisy labels. The FilterNet is trained using the data with clean labels in order to learn the corresponding distribution, increasing loss separability between the data with noisy and clean labels. In the model training phase, the Feature Matcher utilizes the data with both clean and noisy labels to train the TargetNet in the SSL manner, improving the generalization of the global model. In addition, the TargetNet also extracts the class-wise feature embeddings from the data with clean labels. Then, each client uploads its local models and learned feature embeddings to the server. Finally, the server aggregates the local models and updates the global class-wise feature embeddings. This whole process iterates till the model converges.

3.3 *Loss Transformer*

The Loss Transformer is designed to provide unbiased evaluation of the data samples within a client. In centralized learning, the empirical loss has been widely utilized as a simple but effective metric to differentiate clean and noisy samples [13, 18]. However, it cannot be directly applied in the FL scenario for the following reasons. Firstly, the limited amount of local training data within a single client can cause the local DNNs to quickly overfit the data with noisy labels, resulting in the empirical loss having limited discriminatory capability between noisy and clean labels. Secondly, clients are susceptible to confirmation bias, wherein their local models may favor examples that confirm pre-existing beliefs or expectations due to its overfitting, even if those examples contain noisy labels. To demonstrate the impact, we directly apply the selection methodology based on empirical loss [18] to each client in FL. Figure 3 represents the validation result. From Figure 3(a), we observe that the F1 score of each client decreases as the training progresses when the local model is used for prediction. This decrease can be attributed to the clients unhesitatingly selecting the data with noisy labels for training. When the global model is used, the F1 score of each client is low, indicating that the global model is not able to effectively learn from the data of each client, as shown in Figure 3(b).

In order to effectively overcome the above limitation, we introduce FilterNet in the Loss Transformer to tackle the overfitting caused by limited amount of local training data. Moreover, the inter-client integrated loss is designed in the Loss Transformer to deal with confirmation bias. Figure 2 represents the workflow of Loss Transformer.

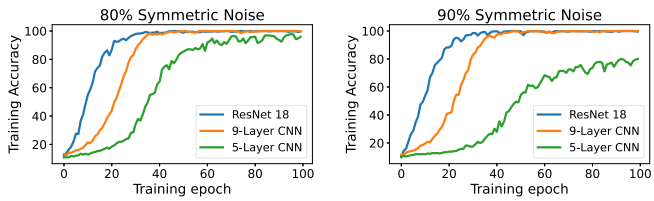
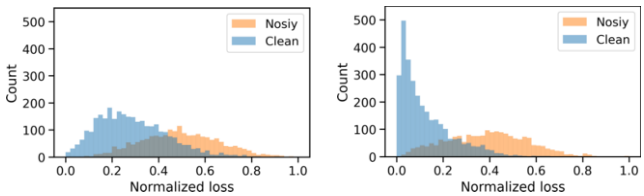


Figure 4. Comparing the convergence speeds of DNNs with varying capacities on the CIFAR10 with symmetric noise. With different capacities of DNNs, we apply the same optimization settings.

FilterNet. FilterNet is designed to prevent the DNN model from quickly overfitting the data with noisy labels, especially on limited data with a high ratio of noise. Directly minimizing the empirical risk on noisy samples can result in the DNNs memorizing all of the corrupt labels during the training process. Although DNNs are prone to overfitting with noisy label samples, the memorization effect can still be leveraged to facilitate learning from clean samples in a simple pattern during the initial stage of training, before gradually memorizing the noisy samples [1]. Based on this effect, various studies [34, 13, 18, 33] have filtered noisy samples based on their loss values. However, these methods overlook the overfitting property of DNNs, which severely decreases the discriminatory capability of empirical loss [1, 36]. Therefore, the problem lies in preventing DNNs from overfitting noisy labeled data quickly, particularly in scenarios where the amount of data is limited and the noise ratio is high. Recently, Cheng et al. [8] theoretically shows that DNNs with lower-capacity can perform better on noisy datasets in centralized learning. Based on this finding, we propose FilterNet with limited capacity in FL. Since it is challenging to design a local model with the appropriate capacity for each client given an arbitrary task, we decouple the DNNs into an encoder (f) with varying amounts of blocks (B) fol-

lowed by a linear classifier (g). The encoder can be further divided into different modules, such as a single convolutional layer or a block of layers (e.g. a residual block). The number of blocks is restricted to be smaller than the TargetNet. This approach allows for flexibility in choosing the appropriate capacity for each client's encoder. Figure 4 compares the convergence rates of DNNs with different capacities under the high ratio of noise. As shown in Figure 4, the shallower network converges slower in limited noisy data with high symmetric noise ratio. This finding suggests that FilterNet is a promising solution to address the quick overfitting problem in FL.



(a) CE from Local model **(b) Inter-client Intergated Loss**

Figure 5. Training on CIFAR-10 with 40% symmetric noise, warm up for 1 global epochs. (a) Standard training with cross-entropy loss from the local FilterNet. (b) Training with inter-client integrated loss.

Inter-client Integrated Loss. Though FilterNet can mitigate the overfitting problem, confirmation bias still severely impacts the sample selection process in FL. In FL, the client's noise ratio is heterogeneous and unknown to the server. For the clients with high ratio of noisy labels, their local models can easily overfit the data with noisy labels, resulting in low empirical risk. In this case, samples with small losses cannot be considered as clean label data since the local models' predictions are biased. The inter-client integrated loss aims to leverage the knowledge of helper clients to improve the prediction reliability of each sample while maintaining privacy. Helper clients should have high data trustability and a similar local model to that of the target client. High data trustability means that the helper clients should have sufficient clean label data to learn the correct relationships between clean data and corresponding clean labels. In FL, the client's data heterogeneity can lead to the divergence of local models among clients. Consequently, measuring the model relevance between the target client and candidate helper clients can enhance model stability. To select H reliable helper clients for the target client k, we design a noisy model discrepancy metric, denoted as DIP,

$$DIP(\theta_F^k, \theta_F^i) = \epsilon_i \sum_{l=1}^{L} \|\theta_{F_l}^k - \theta_{F_l}^i\|^2 \tag{1}$$

where $\epsilon_i = \frac{N_{noise}^i}{N^i}$; N_{noise}^i is the number of data with noisy labels on client i; N^i is the number of local training data on client i; L is the total number of layers in the FilterNet and l is the layer index. It is important to note that the smaller the value of DIP, the more reliable the helper clients are for the target client. During each communication round, the server collects all participating clients' models and the noisy label ratio. The N_{noise}^i is estimated using Equation 7 (detailed discussion in Section 3.5). The server then selects H helper agents with smaller DIP for the target client in the next round.

After determining the H helper clients, we use an integrated loss $\mathcal{L}_{IC}$ to fully exploit the cooperative characteristic of FL, which is defined as

$$\mathcal{L}_{IC}(x_i^k) = \mathcal{L}(x_i^k; \theta_F^k) + \sum_{j=1}^{H} \mathcal{L}(x_i^k; \theta_F^{h_j}) \tag{2}$$

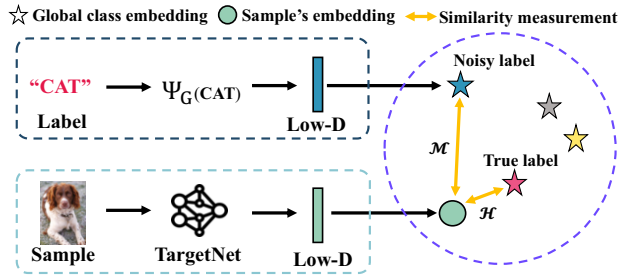


Figure 6. The workflow of Feature Comparator.

Notably, the helper's or client's FilterNet is only used to perform inference in $\mathcal{L}_{IC}$ for data selection. It will not be optimized in the data selection phase. Figure 5 shows the distribution of the inter-client integrated loss, which is more separable than the local client's CE. The data with clean labels have a smaller inter-client integrated loss, which is easier to be modeled by a Gaussian Mixture Model (GMM) [20] model.

To estimate the probability of each local training sample being the clean label data, we employ a two-component GMM [20] to fit $\{\mathcal{L}_{IC}(x_i^k)\}_{i=1}^{N_k}$. For a sample x_i^k of client k, its probability of having a clean label, denoted as $\mathbf{p}(x_i^k)$, is computed using the posterior probability $\mathbf{p}(g|\mathcal{L}_{IC}(x_i^k))$, where g is the Gaussian component with a smaller mean (i.e., smaller loss).

3.4 *Feature Comparator*

In real-world scenarios, label noise can be classified into two types: class-dependent and feature-dependent [30, 1]. The Loss Transformer has demonstrated promising results in improving the ability to detect class-dependent label noise in FL. However, its discriminatory capability may not be as effective in identifying the feature-dependent noise, which is a more complicated form of label noise resulting from human annotation [30]. This has motivated the design of Feature Comparator, which aims to increase the robustness of noisy-label detection in terms of latent representations. Inspired by contrastive representation learning (CRL) [5], we propose to utilize the latent representation information of local samples to discriminate the data with noisy and clean labels. Generally, the representation distance between a sample and the embedding of its corresponding true class should be small, and the distance to the embedding of the incorrect class should be large. Specific to our problem, the main challenges are two folds. First, how to create the correct contrastive pair when the label noise exists? Second, how to discriminate the noisy label and clean label data by contrastive representations?

To create the correct contrastive pairs with data having noisy labels in FL, we introduce the global class-wise feature embedding Ψ_G. In every round, the client k learns the feature ψ_k^c of each class c from the local clean data D_{clean}^k divided by Data Selector (Section 3.5), namely,

$$\psi_k^c = \frac{1}{\hat{n}_k^c} \sum_{i=1}^{\hat{n}_k^c} f_{\theta_T^k}(x_i^k) \mathbb{1}(y_i^k = c) \tag{3}$$

where $f_{\theta_T}(\cdot)$ is the encoder of the TargetNet; $\mathbb{1}(\cdot)$ is the indicator function. The TargetNet is the model we want to train through FL, which is considered to have strong feature extraction ability. Due to data heterogeneity and different noise distributions in clients, the local ψ is not reliable enough for building contrastive pairs. Therefore,

the server collects the learned local features from clients and aggregates them to the global feature embedding Ψ_G^c for class c by taking the average.

To filter out the samples with noisy labels using contrastive representations, we introduce a scheme to utilize the global class feature embedding for comparing the sample and its label. Intuitively, the sample's feature is closest to its corresponding class's global feature embedding. For the sample x_i of the client k, we compare the feature of the sample with all classes' global features and find its closest class,

$$\mathcal{M}(x_i^k) = \text{Max}\big(\{\text{Sim}(f_{\theta_T^k}(x_i^k), \Psi_G(c))\}_{c=1}^C\big) \tag{4}$$

where $\text{Sim}(\cdot,\cdot)$ is the cosine similarity function. Because the sample's closest class estimated by Equation 4 may be wrong, we also compare the extracted feature of the sample with its corresponding class's global feature,

$$\mathcal{H}(x_i^k) = \text{Sim}(f_{\theta_T^k}(x_i^k), \Psi_G(y_i^k)) \tag{5}$$

Finally, we measure each sample of the client k by a feature comparison score $\mathbf{FC}(\cdot,\cdot)$ to filter out the data with noisy labels,

$$\mathbf{FC}(x_i^k, y_i^k) = |\mathcal{M}(x_i^k) - \mathcal{H}(x_i^k)| \tag{6}$$

If the sample's label is noisy, the feature comparison score will be large due to the representation inconsistency. It is important to note that the Feature Comparator differs from the existing similarity-based methods [34]. The comparison mechanism mentioned above is more efficient in capturing the dissimilarity between the clean and the noisy feature distributions in feature-dependent noise.

Discussion. It is worth noting that sharing the global class-wise feature does not lead to privacy leakage [11]. This is because the global class-wise feature is derived from deeper layers and consolidates similar object features multiple times, encompassing even the presence of noisy objects. These consolidated features encompass category-specific information rather than individual sample features. Hence, it does not compromise the confidentiality of the client's samples.

3.5 Data Selector

The Data Selector is designed to separate data with noisy labels and clean labels by measuring the proposed reliability score, which jointly considers the information from Loss Transformer and Feature Comparator.

Specifically, we first warm up the FilterNet for T_{wp} global rounds by training with all clients' data without selection in the standard FL procedure. After T_{wp} rounds, given a client k, each local sample x_i is evaluated by the reliability score function $\mathcal{R}(\cdot)$. At the early stage of training, the quality of learned features by the TargetNet is typically not good enough for detecting the noisy label. Therefore, we dynamically adjust the contribution of feature comparison in the data reliability score function, as shown below,

$$\mathcal{R}(x_i^k) = \mathbf{p}(x_i^k) + \mathbb{1}(T_f < t)\exp(-\mathbf{FC}(x_i^k, y_i^k)) \tag{7}$$

where $\mathbb{1}(T_f > t)$ is the trigger of Feature Comparator, and T_f is a hyperparameter to adjust when the Feature Comparator takes effect. The local sample with the reliability score larger than the threshold $\tau(t)$ is classified as clean-label data and otherwise as noisy-label data. In all the following experiment, $\tau(t)$ is 0.5 for $t \le T_f$, 1 for $t > T_f$ and $T_f = 10$.

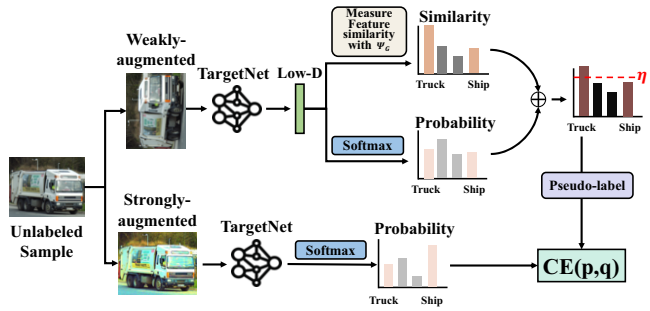


Figure 7. The workflow of Feature Matcher.

3.6 Feature Matcher

After dividing the local data into clean label and noisy label datasets, we simultaneously train the FilterNet and TargetNet, guaranteeing that the two networks perform well on their corresponding components. For the FilterNet, to increase the loss separability between the noisy and clean label data, we perform supervised training with clean label data D_{clean} by minimizing the cross entropy loss $\mathcal{L}(\theta_F)$.

As for the TargetNet, in order to effectively utilize the entire training data, we treat the samples with noisy labels as the unlabeled data and the samples with clean labels as labeled data. For the unlabeled data, we can directly apply the existing semi-supervised learning techniques such as [3] [21] [32]. However, these techniques are designed for centralized learning. In FL, Jeong et al. [15] empirically find that these SSL methods would forget the knowledge learned from the labeled data due to the model inconsistency across multiple clients. Therefore, based on the FixMatch [21], we propose the Feature Matcher, a feature-consistency SSL technique for FL.

To be specific, for the labeled data (i.e., the clean data), we apply the weak augmentation (i.e. the standard flipping and rotation operations) $\alpha(\cdot)$ to them. Then, the cross entropy loss l_s is computed. For the unlabeled data, the Feature Matcher first produces artificial labels on the weakly-augmented unlabeled data. In FL, the aforementioned model inconsistency is caused by the heterogeneous data distributions among different clients. The inconsistency also affects the learning of unlabeled data, resulting in the poor pseudo-labeling on the unlabeled data. To prevent the clients from producing inconsistent pseudo-labeling on unlabeled data, we integrate the global class features Ψ_G into the pseudo-labeling to mitigate the heterogeneity on pseudo-labeling. The global class features can bring the feature-level knowledge to each client, which is beneficial to produce high-quality and consensus pseudo labels. Specifically, given a weakly-augmented unlabeled sample u, we calculate its similarity score $\mathcal{S}$ with the global class feature Ψ_G,

$$\mathcal{S}(\alpha(u)) = \{Sim(f_{\theta_T^k}(\alpha(u)), \Psi_G(c))\}_{c=1}^C \tag{8}$$

Then we obtain the class predicted distribution score q by combining $\mathcal{S}$ and the global model's prediction,

$$q(u) = p_{\theta_T}(\alpha(u)) + \mathcal{S}(\alpha(u)), \quad u \in D_{noise} \tag{9}$$

After that, we use $\hat{q}(u) = \text{argmax}(q(u))$ as a pseudo label of u and calculate the cross-entropy loss with the prediction of unlabeled u which is transformed by the strong augmentation $\mathcal{A}$,

$$\ell_u = \mathbb{1}\Big(\max\big(q(u)\big) \ge \eta\Big)\text{CE}\Big(\hat{q}(u), p_{\theta_T}\big(\mathcal{A}(u)\big)\Big) \tag{10}$$

For the strong augmentation, we select the transformations (i.e., adjusting sharpness, color, solarize, and so forth) with uniform distribution. Finally, the TargetNet is optimized by the following combined

Table 1. The average of last 10 rounds test accuracy (%) on CIFAR-10 and SVHN with IID setting at different noise rates and types. The best accuracy for each noise level is boldfaced.

DataSet	CIFAR-10							SVHN						
Noise Type	Symmetric				Pairflip			Symmetric				Pairflip		
Noisy ratio	0.4	0.5	0.6	0.7	0.25	0.35	0.45	0.4	0.5	0.6	0.7	0.25	0.35	0.45
FedAvg	48.86	37.43	28.82	20.94	60.34	48.54	40.23	59.99	50.01	40.01	30.01	74.99	65.01	54.99
F-Co teaching	83.73	78.90	69.92	54.93	87.03	83.54	67.08	71.36	59.62	45.25	32.65	79.25	65.63	53.75
INCV	55.65	50.94	46.32	24.87	62.39	62.39	40.45	61.45	51.78	39.38	29.25	74.45	62.13	48.53
DS	73.58	71.30	49.76	32.95	81.40	43.75	55.16	66.53	53.22	39.27	49.73	84.60	73.59	55.64
CLC	89.90	87.49	82.09	68.85	89.56	87.05	77.76	93.73	90.10	82.56	34.41	94.52	91.17	70.30
RoFL	87.11	82.93	75.01	61.73	91.53	89.65	86.65	93.51	89.93	81.72	66.59	94.65	85.45	78.07
Ours	**92.03**	**90.52**	**89.30**	**83.72**	**92.83**	**91.75**	**91.67**	**95.53**	**95.04**	**93.47**	**91.23**	**95.91**	**95.73**	**95.12**

loss,

$$\mathcal{L}(\theta_T) = \ell_s + \lambda_u \ell_u \tag{11}$$

where λ_u is to control the weight of the unlabeled loss. In this study, we schedule $\lambda_u = \lambda_{u_0} \min(\frac{t}{T_{ssl}}, 1)$, where t is the global training round and λ_{u_0} is an initial value.

4 Evaluation

4.1 Experimental Setup

Datasets. We validate the effectiveness of FedCoop on representative datasets including SVHN[1], CIFAR-10[2], and CIFAR-10N [30]. In addition, we set up the data distribution and noise situation across different participating clients as follows. We follow [33] to perform the non-IID data partition. Specifically, for a certain class j, it is sampled from the Bernoulli distribution with a fixed probability p to generate the class distribution among all the clients. This class distribution indicates whether the local dataset of client i contains class j. Then the number of training samples belonging to class j in client i is sampled from the symmetric Dirichlet distribution [10] with the parameter α_{DIR} which determines the concentration of Dirichlet. For each client, we inject two types of label noise into its local data [13]: (a) Symmetric Noise [27]: the true label is flipped into the wrong label sampled from the uniform distribution; (b) Pairflip Noise [13]: the original label is only flipped into the similar classes. To simulate heterogeneous noise levels, each participant's noise level ϵ is drawn from Beta Distribution with parameters β and α.

Baselines. Two groups of baselines are adopted. The first group consists of the methods that tackle noisy labels in centralized learning: 1) **INCV** [4] employs cross-validation to identify clean samples and train on them; 2) **Co-teaching** [13] simultaneously trains two DNNs, and each network chooses the instances with the small loss as its peer network's training data. The second group contains the methods designed to tackle label noise in FL: 1) **DS** [26] assumes the server contains clean benchmark data and selects the client's local data that are relevant to the benchmark data to train the local model; 2) **RoFL** [34] shares the central representation of clients' local data to maintain the constant decision boundary over clients and performs label correction on data with noisy labels; 3) **CLC** [35] identifies the local data with noisy labels by consensus-defined class-wise information and performs label correction on data with noisy labels.

Implementation Details. For the experiments on CIFAR-10 and SVHN, we use a 9-layers CNN [13] as FilterNet and an 18-layer ResNet [14] as TargetNet. We adopt stochastic gradient descents (SGD) as the optimizer for all the experiments, with the momentum of 0.9, weight decay of 0.0005, and a learning rate of 0.01. We set the number of local epochs as 10, the local batch size as 64, and the number of global epochs as 200.

[1] http://ufldl.stanford.edu/housenumbers/
[2] https://www.cs.toronto.edu/ kriz/cifar.html

Table 2. Average of last 10 rounds test accuracy (%) of different methods on SVHN and CIFAR-10 with different noise distribution settings of symmetric noise.

Dataset	CIFAR-10			SVHN		
Beta (α, β)	$(3, 4)$	$(2, 3)$	$(7, 5)$	$(3, 4)$	$(2, 3)$	$(7, 5)$
Noise Rate	0.2 – 0.7	0.3 – 0.6	0.4 – 0.7	0.2 – 0.7	0.3 – 0.6	0.4 – 0.7
FedAvg	43.63	47.90	32.52	59.03	65.07	47.01
F-Co teaching	78.23	78.53	78.63	68.19	76.06	54.80
INCV	46.05	49.12	36.54	59.04	50.77	35.59
DS	70.31	74.74	51.23	81.93	82.21	71.82
CLC	88.56	88.71	84.84	92.88	93.72	90.57
RoFL	87.02	88.72	83.34	92.45	92.21	82.87
Ours	**91.60**	**91.66**	**90.16**	**95.12**	**95.64**	**94.50**

Table 3. Average of last 10 rounds test accuracy (%) on CIFAR-10 with different non-IID and noise settings.

Non-IID Type	(α_{DIR} = 1000, p = 0.9)			(α_{DIR} = 100, p = 0.9)		
Noise Type	Symmetric		Pairflip	Symmetric		Pairflip
Noise Rate	0.4	0.5	0.35	0.4	0.5	0.35
FedAvg	40.32	32.84	44.75	41.70	33.54	45.66
F-Co teaching	71.68	61.14	71.63	72.14	59.59	69.46
INCV	49.29	39.41	52.82	47.12	40.15	53.34
DS	64.54	51.51	69.91	65.11	52.34	68.78
CLC	86.32	82.85	83.32	86.25	81.13	80.90
RoFL	86.71	81.50	88.72	86.01	78.77	88.11
Ours	**92.01**	**90.94**	**92.62**	**91.68**	**90.51**	**92.19**

4.2 Evaluation of Robustness

Different Ratios and Various Types of Noisy Data. We first evaluate the robustness of FedCoop to different types of noise. Table 1 shows the model performance with different methods for two types of noise (*symmetric* and *pairflip*) with different ratios. FedCoop achieves the best performance on both types of noise. For symmetric and pairflip noise in CIFAR-10, FedCoop improves the model accuracy at most by 62.78%, 51.44%, 32.59%, and on average 27.89%, 22.38%, respectively, compared with FedAvg. In addition, the performance of all the baselines prominently decays with the increase of noise ratio. However, FedCoop maintains high robustness under different noise scenarios for the reason that the Loss Transformer and Feature Comparator provide high-quality data selection on each client. Especially, for the extremely symmetric noise (noise ratio > 60%) on CIFAR-10, FedCoop outperforms the baselines with a larger margin: 62.78% over FedAvg, 28.79% over Co-teaching, 58.85% over INCV, 21.99% over RoFL, 44.07% over DS and 21.59% over CLC. This is because the Feature Matcher effectively improves the data utilization, making the model intelligently learn from the noisy labels to increase its generalizability. Moreover, Table 2 shows the model performance with different distributions of noisy labels, demonstrating the effectiveness and stability of FedCoop under various scenarios.

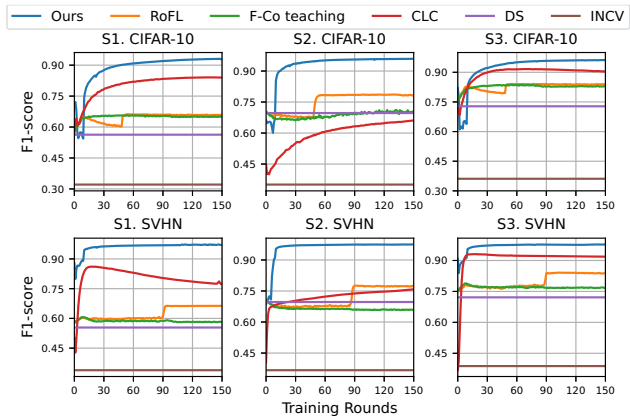


Figure 8. Comparison of F1-score on CIFAR-10 and SVHN with different label noise settings. S1 denotes the 60% symmetric noise on each client. S2 denotes the 45% pairflip noise on each client. S3 denotes the clients' heterogeneous noise ratio modeling by beta distribution ($\alpha = 3, \beta = 4$).

Heterogeneous Data Distribution. In this part, we further evaluate the effectiveness of FedCoop in non-IID settings. We follow the non-IID data partition and inject different types and ratios of noise into the data on each client. Table 3 represents the result on CIFAR-10 with different settings. FedCoop consistently outperforms the baselines by 5.30% to 58.10%. Compared with the same noise setting in the IID data partition, all the baselines have performance degradation and the worst one degrades by 37.76%. This is because the data distribution and label noise jointly impact the performance of local models at the same time. On the contrary, FedCoop is still effective in non-IID settings. The Feature Matcher exploits the global class feature to produce the high-consistency pseudo-label across the clients in order to further improve the generalization of the global model. Thus, the global model can still learn well on clients with different data distributions.

Real-world Human Annotated Noise. In order to evaluate FedCoop in real-world noise scenarios, we adopt the CIFAR-10N [30] dataset in this experiment with the IID setting. The CIFAR-10N is a dataset annotated by human annotators with different background and knowledge discrepancies. Compared with synthetic noise, the noise transition matrix of CIFAR-10N is complex and hard to model. Moreover, the pattern of noise labels in CIFAR-10N is feature-dependent [30] instead of class-dependent, which is challenging to model and predict the noise distribution since the noise transition matrices are complex. Table 4 shows the best accuracy of different methods on CIFAR-10N. FedCoop outperforms other methods by at most 39.61%, and 17.49% on average. The feature-dependent noise easily confuses the loss, leading the loss-based methods to fail. However, the Feature Comparator of FedCoop utilizes the global class feature to detect the noisy label, effectively keeping the local model from being corrupted by noise data. These evidences show that FedCoop is effective for the real-world human annotated noisy label dataset.

Table 4. Best test accuracy on CIFAR-10N (Worst) with IID setting.

Methods	FedAvg	F-Co teaching	INCV	DS	CLC	RoFL	Ours
Acc.(%)	81.07	81.18	54.40	74.71	84.19	83.57	**94.01**

4.3 Analysis on Sample Selection

The effectiveness of sample selection is essential for training a robust model. Hence, we use F1-score as the metric to evaluate the effectiveness of sample selection. F1 score is widely used in noisy label detection in centralized learning [17, 7]. Figure 8 represents the F1 scores of all the methods on different datasets with various noise types and distributions in the IID setting. CLC and RoFL are trapped in the confirmation bias, leading to performance degradation in data selection. It shows that the centralized methods are not applicable in FL scenario. The F1 score of the FedCoop is consistently higher than other baselines under different ratios of noisy labels during the training process. This is because FedCoop utilizes the helper clients to effectively mitigate the confirmation bias caused by a single client. Compared with the methods that only utilize the sample's loss to filter out the data with noisy labels, FedCoop identifies the data with noisy labels in terms of feature space, which is more effective and stable in selecting the data with clean labels with the heterogeneous distribution. Hence, FedCoop can provide a high-quality sample selection without losing the essential information from the clean label data, making the local model effectively learn from clean data.

Table 5. Ablation study results (average of last 10 rounds test accuracy (%)) on CIFAR-10 with different noise settings.

Noise Type	Symmetric			Pairflip
Noise Rate	0.4	0.7	0.4 with non-IID ($\alpha_{DIR} = 100, p = 0.9$)	0.35
Ours	**92.03**	**83.72**	**91.68**	**91.75**
Ours w/o Helper Clients	89.89	63.79	88.55	88.41
Ours w/o FilterNet	90.42	66.87	89.23	90.22
Ours w/o Feature Comparator	85.95	79.62	66.86	89.23
Ours w/o Feature Matcher	72.93	37.33	74.37	75.98

Table 6. Average of last 10 rounds test accuracy (%) of on CIFAR-10 using FedCoop with different number of helper clients.

Noise Type	Symmetric			Paitflip
H	0.4	0.2 – 0.7	0.4 with non-IID ($\alpha_{DIR} = 100, p = 0.9$)	0.35
1	91.08	90.20	90.73	91.67
2	91.23	90.37	91.21	91.87
3	90.70	89.00	90.70	91.36
4	90.07	89.04	90.61	91.63
5	91.04	89.35	90.74	91.43

4.4 Ablation and Sensitivity Study

Ablation Study. To study the effectiveness of each component of FedCoop, we conduct the ablation study with four different noise scenarios. Table 5 shows the corresponding result. We can observe that Feature Matcher has critical contribution to FedCoop. It represents that only using the clean label data is not effective for training the global model, especially when the noise ratio is high. In the non-IID setting, the Feature Comparator plays an important role in data selection since the accuracy dropped at most 24.82% compared with other components. This indicates that only relying on the instance's loss to filter data with noise labels out is not robust in the heterogeneous data distribution. The Feature Matcher also significantly benefits in non-IID data partition settings since the knowledge of the global feature can improve the consistency of pseudo-labeling across the clients, which mitigates knowledge forgetting in FL. For the Loss Transformer, the FilrerNet and the cooperation of helper clients significantly contribute to the effectiveness of FedCoop in the heavy noise scenario. This shows that the proposed collaboratively data selection mechanism is reasonable and essential.

Sensitivity Study of H. The number of helper clients H impacts the communication cost and the performance of the global model at the same time. Therefore, we conduct experiments to investigate the performance variation as more helper clients are selected for the Loss Transformer. We set the total number of clients as 20 with different noise settings. Table 6 shows the performance of global model

on CIFAR-10 with the different number of helper clients. The result indicates that the small number ($H \leq 2$) of noisy clients is sufficient to achieve high accuracy regardless of different noise levels. We can also observe that increasing the number of helper clients only has a little improvement on the global model's performance ($< 1\%$). Therefore, choosing one or two helper clients can already guarantee the robustness and effectiveness of the FedCoop. It can cause negligible extra communication overhead at the same time.

5 Conclusion

In this paper, we present FedCoop, a feature comparison-based cooperative federated learning framework for noisy labels. To address the heterogonous noise distribution on the clients, the FedCoop selects the clean label data by combining the evaluation from the Loss Transformer and Feature Comparator. The Loss Transformer utilizes the FilterNet to filter the noisy label based on the inter-client integrated loss that aims to eliminate the confirmation-bias problem from the local model. In addition, the Feature Comparator detects the noisy label by comparing the sample's feature with the global class feature. After selecting the clean label data, the Feature Matcher performs the adaptive SSL on the TargetNet to train the robust global model. Systematic evaluations have demonstrated the effectiveness of the proposed FedCoop.

6 Acknowledgements

This paper is supported the Science and Technology Development Fund of Macau SAR (File no. 0021/2022/ITP, 0081/2022/A2, SKL-IOTSC(UM)-2021-2023, 0123/2022/AFJ, and 0015/2019/AKP), GuangDong Basic and Applied Basic Research Foundation (No. 2020B515130004), and Key-Area Research and Development Program of Guangdong Province (No. 2020B010164003), and SRG2022-00010-IOTSC.

References

[1] Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al., 'A closer look at memorization in deep networks', in *International conference on machine learning*, pp. 233–242. PMLR, (2017). 1, 3, 4

[2] Michele Banko and Eric Brill, 'Scaling to very very large corpora for natural language disambiguation', in *Proceedings of the 39th annual meeting of the Association for Computational Linguistics*, pp. 26–33, (2001). 1

[3] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel, 'Mixmatch: A holistic approach to semi-supervised learning', *Advances in neural information processing systems*, **32**, (2019). 5

[4] Pengfei Chen, Ben Ben Liao, Guangyong Chen, and Shengyu Zhang, 'Understanding and utilizing deep neural networks trained with noisy labels', in *International Conference on Machine Learning*, pp. 1062–1070. PMLR, (2019). 2, 6

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton, 'A simple framework for contrastive learning of visual representations', in *International conference on machine learning*, pp. 1597–1607. PMLR, (2020). 4

[6] Yiqiang Chen, Xiaodong Yang, Xin Qin, Han Yu, Biao Chen, and Zhiqi Shen, 'Focus: Dealing with label quality disparity in federated learning', *arXiv preprint arXiv:2001.11359*, (2020). 1, 2

[7] Hao Cheng, Zhaowei Zhu, Xingyu Li, Yifei Gong, Xing Sun, and Yang Liu, 'Learning with instance-dependent label noise: A sample sieve approach', *arXiv preprint arXiv:2010.02347*, (2020). 7

[8] Hao Cheng, Zhaowei Zhu, Xing Sun, and Yang Liu, 'Mitigating memorization of noisy labels via regularization between representations', in *The Eleventh International Conference on Learning Representations*, (2023). 3

[9] Xiuwen Fang and Mang Ye, 'Robust federated learning with noisy and heterogeneous clients', in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10072–10081, (2022). 2

[10] Bela A Frigyik, Amol Kapila, and Maya R Gupta, 'Introduction to the dirichlet distribution and related processes', *Department of Electrical Engineering, University of Washignton, UWEETR-2010-0006*, (0006), 1–27, (2010). 6

[11] Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller, 'Inverting gradients-how easy is it to break privacy in federated learning?', *Advances in Neural Information Processing Systems*, **33**, 16937–16947, (2020). 5

[12] Aritra Ghosh, Himanshu Kumar, and PS Sastry, 'Robust loss functions under label noise for deep neural networks', in *Proceedings of the AAAI conference on artificial intelligence*, volume 31, (2017). 1, 2

[13] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama, 'Co-teaching: Robust training of deep neural networks with extremely noisy labels', *Advances in neural information processing systems*, **31**, (2018). 1, 2, 3, 6

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, 'Deep residual learning for image recognition', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, (2016). 6

[15] Wonyong Jeong, Jaehong Yoon, Eunho Yang, and Sung Ju Hwang, 'Federated semi-supervised learning with inter-client consistency & disjoint learning', *arXiv preprint arXiv:2006.12097*, (2020). 5

[16] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al., 'Advances and open problems in federated learning', *Foundations and Trends® in Machine Learning*, **14**(1–2), 1–210, (2021). 1

[17] Taehyeon Kim, Jongwoo Ko, JinHwan Choi, Se-Young Yun, et al., 'Fine samples for learning with noisy labels', *Advances in Neural Information Processing Systems*, **34**, (2021). 7

[18] Junnan Li, Richard Socher, and Steven CH Hoi, 'Dividemix: Learning with noisy labels as semi-supervised learning', *arXiv preprint arXiv:2002.07394*, (2020). 2, 3

[19] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith, 'Federated optimization in heterogeneous networks', *Proceedings of Machine Learning and Systems*, **2**, 429–450, (2020). 2

[20] Haim Permuter, Joseph Francos, and Ian Jermyn, 'A study of gaussian mixture models of color and texture features for image classification and segmentation', *Pattern recognition*, **39**(4), 695–706, (2006). 4

[21] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li, 'Fixmatch: Simplifying semi-supervised learning with consistency and confidence', *Advances in Neural Information Processing Systems*, **33**, 596–608, (2020). 5

[22] Kahou Tam, Li Li, Bo Han, Chengzhong Xu, and Huazhu Fu, 'Federated noisy client learning', *arXiv preprint arXiv:2106.13239*, (2021). 1

[23] Daiki Tanaka, Daiki Ikami, Toshihiko Yamasaki, and Kiyoharu Aizawa, 'Joint optimization framework for learning with noisy labels', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5552–5560, (2018). 1, 2

[24] A Tarvainen and H Valpola, 'Weight-averaged consistency targets improve semi-supervised deep learning results. corr abs/1703.01780', *arXiv preprint arXiv:1703.01780*, **1**(5), (2017). 1

[25] Chunlin Tian, Li Li, Zhan Shi, Jun Wang, and ChengZhong Xu, 'Harmony: Heterogeneity-aware hierarchical management for federated learning system', in *2022 55th IEEE/ACM International Symposium on Microarchitecture (MICRO)*, pp. 631–645. IEEE, (2022). 1

[26] Tiffany Tuor, Shiqiang Wang, Bong Jun Ko, Changchang Liu, and Kin K Leung, 'Overcoming noisy and irrelevant data in federated learning', in *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 5020–5027. IEEE, (2021). 1, 2, 6

[27] Brendan Van Rooyen, Aditya Menon, and Robert C Williamson, 'Learning with symmetric label noise: The importance of being unhinged', *Advances in neural information processing systems*, **28**, (2015). 6

[28] Andreas Veit, Neil Alldrin, Gal Chechik, Ivan Krasin, Abhinav Gupta, and Serge Belongie, 'Learning from noisy large-scale datasets with minimal supervision', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 839–847, (2017). 1, 2

[29] Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey, 'Symmetric cross entropy for robust learning with noisy labels', in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 322–330, (2019). 1, 2

[30] Jiaheng Wei, Zhaowei Zhu, Hao Cheng, Tongliang Liu, Gang Niu, and Yang Liu, 'Learning with noisy labels revisited: A study using real-world human annotations', in *International Conference on Learning Representations*, (2022). 1, 4, 6, 7

[31] Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang, 'Learning from massive noisy labeled data for image classification', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2691–2699, (2015). 1

[32] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le, 'Unsupervised data augmentation for consistency training', *Advances in Neural Information Processing Systems*, **33**, 6256–6268, (2020). 5

[33] Jingyi Xu, Zihan Chen, Tony QS Quek, and Kai Fong Ernest Chong, 'Fedcorr: Multi-stage federated learning for label noise correction', *arXiv preprint arXiv:2204.04677*, (2022). 1, 2, 3, 6

[34] Seunghan Yang, Hyoungseob Park, Junyoung Byun, and Changick Kim, 'Robust federated learning with noisy labels', *IEEE Intelligent Systems*, (2022). 1, 2, 3, 5, 6

[35] Bixiao Zeng, Xiaodong Yang, Yiqiang Chen, Hanchao Yu, and Yingwei Zhang, 'Clc: A consensus-based label correction approach in federated learning', *ACM Transactions on Intelligent Systems and Technology (TIST)*, (2022). 1, 2, 6

[36] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals, 'Understanding deep learning (still) requires rethinking generalization', *Communications of the ACM*, **64**(3), 107–115, (2021). 3