Federated Learning with Extremely Noisy Clients via Negative Distillation

Yang Lu [1,2], Lin Chen [1,2], Yonggang Zhang [3], Yiliang Zhang [1,2], Bo Han [3], Yiu-ming Cheung [3], Hanzi Wang [1,2*]

[1]Fujian Key Laboratory of Sensing and Computing for Smart City, School of Informatics, Xiamen University, China
[2]Key Laboratory of Multimedia Trusted Perception and Efficient Computing, Ministry of Education of China, Xiamen University, China
[3]Department of Computer Science, Hong Kong Baptist University, Hong Kong, China
luyang@xmu.edu.cn, chenlin191209@163.com, ylzhangcs@hotmail.com, hanzi.wang@xmu.edu.cn, csygzhang@comp.hkbu.edu.hk, bhanml@comp.hkbu.edu.hk, ymc@comp.hkbu.edu.hk

Abstract

Federated learning (FL) has shown remarkable success in cooperatively training deep models, while typically struggling with noisy labels. Advanced works propose to tackle label noise by a re-weighting strategy with a strong assumption, i.e., mild label noise. However, it may be violated in many real-world FL scenarios because of highly contaminated clients, resulting in extreme noise ratios, e.g., $>$90%. To tackle extremely noisy clients, we study the robustness of the re-weighting strategy, showing a pessimistic conclusion: minimizing the weight of clients trained over noisy data outperforms re-weighting strategies. To leverage models trained on noisy clients, we propose a novel approach, called *ne*gative *d*istillation (FedNed). FedNed first identifies noisy clients and employs rather than discards the noisy clients in a knowledge distillation manner. In particular, clients identified as noisy ones are required to train models using noisy labels and pseudo-labels obtained by global models. The model trained on noisy labels serves as a 'bad teacher' in knowledge distillation, aiming to decrease the risk of providing incorrect information. Meanwhile, the model trained on pseudo-labels is involved in model aggregation if not identified as a noisy client. Consequently, through pseudo-labeling, FedNed gradually increases the trustworthiness of models trained on noisy clients, while leveraging all clients for model aggregation through negative distillation. To verify the efficacy of FedNed, we conduct extensive experiments under various settings, demonstrating that FedNed can consistently outperform baselines and achieve state-of-the-art performance.

Introduction

The rise of federated learning (FL) benefits from its capacity for large-scale distributed model training in a data-preserving manner (Kairouz et al. 2021). The server aggregates client models to produce a global model and sends it back for subsequent training. When the sample annotation is accurate, the global model can generally exhibit promising performance, even when the data is somehow non-IID distributed (Ma et al. 2022). Another challenge in FL is the label-noise problem. Usually, as each client collects and annotates the data by itself, the inaccurate annotation in each client may be with different degrees (Xu et al. 2022). Different from the label-noise learning in a batch setting, the server in FL needs to judge the degree of label noise for each client before model aggregation, because the server has no information about which client has label noise.

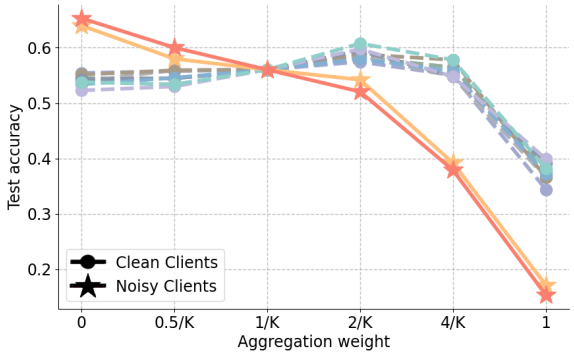


Figure 1: The test accuracy of the global model by controlling the weight of a single client model. We set ten client models including eight clean ones (with a noise ratio of 0%) and two extremely noisy ones (with a noise ratio of 99%). K is the total number of clients, which is ten in this example.

Many efforts have been devoted to alleviating the label-noise issue in FL (Liang et al. 2023; Yang et al. 2021; Chen et al. 2020; Wang et al. 2022; Yang et al. 2022; Kim et al. 2022; Xu et al. 2022; Wu et al. 2023a). Advanced works have shown the promising benefits of assigning different weights to each client during the model aggregation process. The intuition behind these methods is that all client models have the potential to contribute to the global model by model aggregation, highlighting the importance of aggregation weights assigned for each client. The basic intuition is built upon a strong assumption that the label noise on each client is relatively mild. Namely, models trained on mild noise can benefit global models by aggregation.

However, the strong assumption could be violated in many practical FL scenarios. For instance, some clients may be highly contaminated with noise ratios exceeding 90%, due to unintentional mislabeling or deliberate data poisoning. These clients are referred to as 'extremely noisy' clients in the context of this work. Consequently, models trained

*Corresponding author.

on extremely noisy clients may perform differently on the same dataset, causing global models to degrade via model aggregation. We assign different weights used for aggregation to illustrate the negative impacts of models trained on extremely noisy clients, as depicted in Figure 1. These experiments show that the performance of the global model varies with the weight assigned to a client model. Specifically, for each line plot, we merely change the weight for one client model, while keeping weights for the left clients equally [1]. Our results show that a) models trained on clean clients contribute to the global model, and b) models trained on extremely noisy clients lead to severe performance degradation. Namely, we should discard models trained on noisy clients, i.e., assigning 0 weights to these models rather than weighing them with an arbitrarily small weight. Therefore, discarding noisy clients is preferred over re-weighting clients for the model aggregation process. However, the discarding strategy goes against the intention of FL.

In this work, we propose Federated learning via Negative distillation (FedNed) to deal with the extreme-noise problem. FedNed first identifies the client models with extreme label noise by model prediction uncertainty (Gal and Ghahramani 2016), since uncertainty is widely used to measure whether a model can be trusted (Jiang et al. 2018). Then, rather than directly discarding them, FedNed utilizes them through a novel strategy called negative distillation. In negative distillation, these client models trained on extremely noisy data act as 'bad teachers' when updating the global model. FedNed keeps the global model's prediction different from that of the extremely noisy client models, which shares the same spirit with negative learning (Kim et al. 2019), i.e., reducing the risk of providing incorrect information. As a result, negative distillation produces an even better global model than the one aggregated by only using client models. Extensive experiments verify the effectiveness of the proposed method on the environment of clients with extremely noisy-labeled data.

Our main contributions are summarized as follows:

- We reveal the severe impacts induced by extremely noisy clients, posing challenges to existing methods. Specifically, involving models trained on extremely noisy clients causes performance degradation of global models.
- We propose a novel method called FedNed to tackle FL with extremely noisy clients. In FedNed, the key idea called negative distillation is proposed to encourage the global model's prediction to be dissimilar to that of noisy models. A new local optimization strategy is subsequently adopted for identified extremely noisy clients.
- We conduct comprehensive experiments to verify the efficacy of FedNed on benchmarks with extremely noisy clients, demonstrating that FedNed significantly and consistently outperforms state-of-the-art methods.

Related Work

We first summarize the advancements achieved in the domains of federated learning (FL) and label-noise learning. Then, we summarize the recent work of the joint problem: label-noise learning in the FL environment.

Federated Learning with Data Heterogeneity

Since the seminal work FedAvg (McMahan et al. 2017) was proposed, the landscape of FL research has predominantly revolved around addressing challenges of data heterogeneity (Ma et al. 2022), where data distributions shift with clients, i.e., non-IID data.

Advanced works have achieved outstanding improvements through various approaches, under an assumption that client data are noise free. FedProx (Li et al. 2020) introduces a regularization mechanism into the local training process by employing a proximal term, effectively enhancing convergence behavior. SCAFFOLD (Karimireddy et al. 2020) mitigates client drift by incorporating supplementary control variates, ensuring stable convergence. FedDyn (Acar et al. 2021) dynamically regulates the training process of neural network models across devices, allowing for efficient training while remaining robust to diverse scenarios. Recently, FedNH (Dai et al. 2023) marks an important stride by enhancing the efficacy of local models in both personalization and generalization. It is achieved through the incorporation of uniformity and class semantics in class prototypes, thereby improving overall model stability and effectiveness across diverse clients. FedNP (Wu et al. 2023b) efficiently estimates the inaccessible ground-truth global data distribution using a probabilistic neural network, mitigating performance degradation induced by data heterogeneity. Advanced works propose to share privacy-free data among clients to tackle data heterogeneity (Tang et al. 2022; Yang et al. 2023), achieving promising performance.

Label-Noise Learning

In numerous real-world scenarios, data annotation often gives rise to the challenge of noisy labels. A considerable number of methods for label-noise learning can be categorized into the following groups (Song et al. 2022): sample selection (Yao et al. 2021; Karim et al. 2022), loss function adjustment (Ghosh, Kumar, and Sastry 2017; Shu et al. 2019), regularization (Xia et al. 2020; Lukasik et al. 2020), and robust model architecture (Han et al. 2018b,a). Among these works, the line of sample selection is the most related approach, as we perform model selection to defy label noise.

Early methods primarily rely on the trick of small risk (or loss), sharing the same spirit with model selection using uncertainty (Jiang et al. 2018). For instance, co-teaching (Han et al. 2018b) adopts sample selection by two distinct models, wherein the clean samples selected by one model are used to train another. Similarly, DivideMix (Li, Socher, and Hoi 2019) effectively employs the small loss trick to select clean samples, subsequently integrating semi-supervised learning by treating the unselected samples as unlabeled. Recently, contrastive learning approaches have been involved in sample selection. Jo-SRC (Yao et al. 2021) utilizes contrastive learning to estimate the likelihood of sample cleanliness or out-of-distribution by training the network with dual predictions and introducing a joint loss with consistency regularization to improve model generalization. Unicon (Karim

[1] The sum of all weights is 1.

et al. 2022) employs a uniform selection mechanism, coupled with contrastive learning, to tackle imbalanced sample selection and prevent the memorization of noisy labels.

Federated Learning on Noisy-Labeled Data

The management of noisy-labeled local data from diverse clients poses a novel challenge within the field of FL (Liang et al. 2023). Specifically, advanced FL methods for data heterogeneity typically struggle with noisy labels, while traditional label-noise learning approaches are no longer robust when facing distributed data.

One straightforward approach is to reserve clean data on the server to identify noisy clients. Among these methods, quantifying each client's noise ratio is used to identify low-noise ratio clients (Yang et al. 2021). The server can subsequently aggregate the client models based on the ranking of estimated noise ratios. Similarly, FOCUS (Chen et al. 2020) assigns different weights to clients based on the credibility of their local data. However, these methods make a strong assumption that the server holds clean labeled data.

Advanced works make a great attempt to weaken the assumption when estimating noise rate. For instance, FedNoiL (Wang et al. 2022) utilizes prediction confidence to estimate the noise ratio, which is then normalized to weight client models during model aggregation. FedCorr (Xu et al. 2022) adopts local intrinsic dimension to differentiate between clean and noisy clients, utilizing the estimated noise rate as a regularization coefficient to constrain model updates. FedNoRo (Wu et al. 2023a) initially identifies noisy clients through normalized local losses, followed by distance-aware model aggregation in the second stage. RoFL (Yang et al. 2022) assesses distances between prototypes to facilitate model aggregation. FedRN (Kim et al. 2022) identifies some reliable neighbors and employs their mixture model to enhance clean sample selection. However, the intuition behind these methods is that all client models can contribute to the global model, which may no longer hold under extremely noisy scenarios.

Proposed Method

In this section, we detail how the proposed method FedNed endows FL with robustness against extremely noisy clients.

Problem Definition

In the typical FL environment (McMahan et al. 2017), a collection of K clients collaborates with a central server to train a global model. Each client collects its local dataset, denoted as $\mathcal{D}_k$, which may contain noisy-labeled samples with an unknown noise ratio. Regarding the noise ratio, existing works assume that it would never surpass a certain threshold, i.e., mild label noise. Many practical scenarios highlight the importance and urgency of relaxing the strong assumption to that of extreme label noise.

In our work, the noise ratio ranges from 0% to 100%, where clients are categorized as 'extremely noisy clients' if their uncertainties exceed a certain threshold. Our focus is mainly on these extremely noisy clients, since this complex scenario is rarely discussed in the literature.

Algorithm Overview

The overall training process of FedNed is illustrated in Figure 2. The server identifies the extremely noisy clients in each round (c.f. Section Identification of Extremely Noisy Clients) and excludes their uploaded models during model aggregation. Subsequently, a novel negative distillation procedure (c.f. Section Negative Distillation) is employed to further enhance the global model's performance by incorporating information from the extremely noisy models. On the server, we utilize a public dataset $\mathcal{D}_U$ for both the identification step and the negative distillation step. To address the privacy issue in FL, $\mathcal{D}_U$ can be a public dataset without label annotation. Local training on each client also takes into account the identification result on the server in the previous round. If a client is identified as extremely noisy, an additional local model is updated and uploaded (c.f. Section Client-Side Training).

Identification of Extremely Noisy Clients

The first step is to identify the client models trained with extreme label noise. In each communication round, the server randomly selects a subset of clients denoted as $\mathcal{A}^t$. Subsequently, the selected clients are divided into two categories: mildly noisy (MN) clients indexed by $\mathcal{C}^t$ and extremely noisy (EN) clients indexed by $\mathcal{N}^t$. EN clients are updated on local extremely noisy-labeled data.

Following previous work (Jiang et al. 2018), we employ model uncertainty (Gal and Ghahramani 2016) to identify models with high risk, i.e., trained with noisy-labeled data. Specifically, client models with high uncertainty are regarded as EN clients. Denoting $\mathbf{w}_k^t$ as the parameter of local model at client k for the t-th round, the probability of model for class c can be calculated by:

$$p(y=c|\mathbf{x},\mathcal{D}_k) := \int p(y=c|\mathbf{x},\mathbf{w})p(\mathbf{w}|\mathcal{D}_k)d\mathbf{w} \quad (1)$$

$$\approx \frac{1}{T}\sum_{t=1}^{T} p(y=c|\mathbf{x},\widehat{\mathbf{w}}_t), \quad (2)$$

where $\mathbf{x} \in \mathcal{D}_U$ is input data with its label y, $p(y=c|\mathbf{x},\mathbf{w})$ stands for the probability label c predicted by model with parameter $\mathbf{w}$, $p(\mathbf{w}|\mathcal{D}_k)$ represents the distribution of applying Dropout operation to models trained on local data $\mathcal{D}_k$ during inference, T is the times to perform inference for each sample, and $\widehat{\mathbf{w}}_t$ denotes parameters sampled from $p(\mathbf{w}|\mathcal{D}_k)$. Built upon $p(y=c|\mathbf{x},\mathcal{D}_k)$, we calculate the uncertainty U_k for client k by averaging over all samples and classes:

$$\frac{-1}{C\cdot|\mathcal{D}_U|}\sum_{c=1}^{C}\sum_{\mathbf{x}\in D_U} p(y=c|\mathbf{x},\mathcal{D}_k)\log p(y=c|\mathbf{x},\mathcal{D}_k). \quad (3)$$

We select the clients whose uncertainty U_k is greater than a threshold λ as the EN clients.

It is worthwhile to note that client models identified as noisy are used to perform model aggregation in previous works (Wu et al. 2023a; Wang et al. 2022), while we identify noisy clients to perform negative distillation.

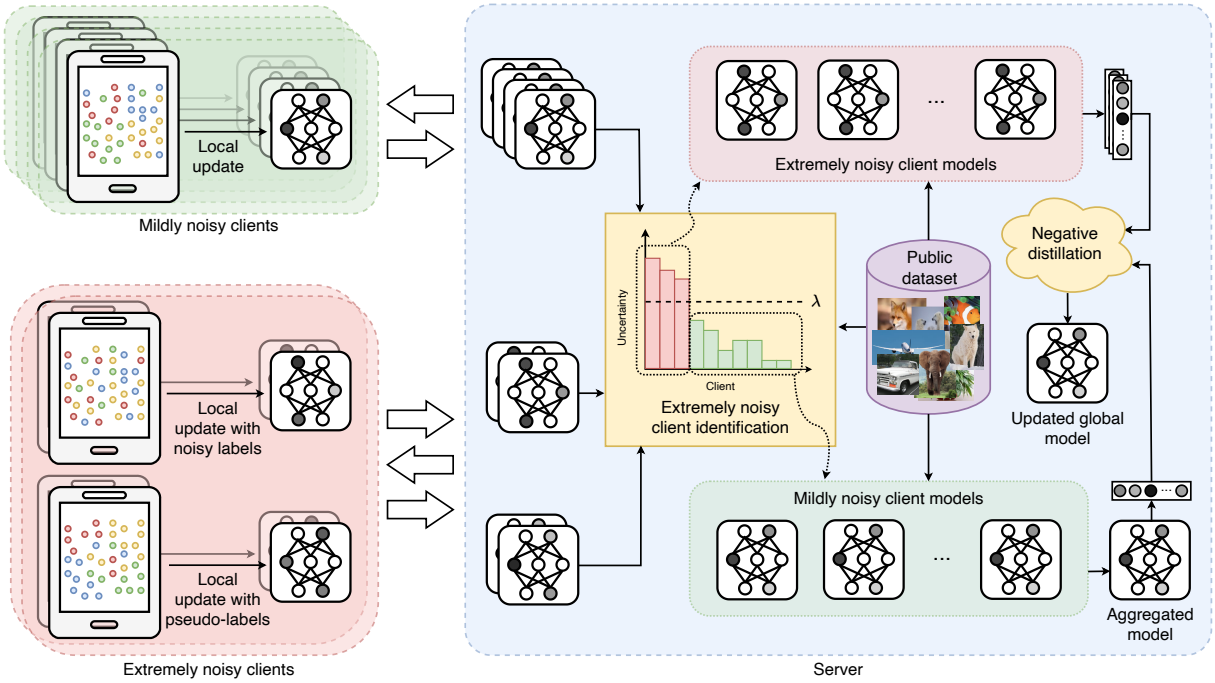


Figure 2: The architecture overview of the proposed FedNed. In each round, the server identifies the mildly noisy (MN) and extremely noisy (EN) client models via MC dropout and prediction uncertainty. Negative distillation is then utilized to incorporate EN client models for a better global model.

Negative Distillation

Building upon the observation and analysis presented in Figure 1, the integration of EN clients into model aggregation with arbitrary weights emerges as detrimental to the generalization performance of the global model. Opting for a straightforward solution that involves discarding the EN clients offers a protective measure to maintain the integrity of the global model. However, these clients could potentially hold information beneficial to enhance the global model. Apparently, the EN clients are trained on the local datasets with extremely noisy labels, leading to their diminished capacity for accurate predictions. Thus, leveraging the incorrect predictions generated by EN clients may compel the global model to diverge from their predictions. Avoiding incorrect predictions is widely used in negative learning, which aims to reduce the risk of providing incorrect information. In this paper, we implement this idea by the concept of negative distillation, making the global model's prediction diverge from those offered by the identified EN clients. Negative distillation is utilized to incorporate EN client models for a better global model.

Inspired by FedDF (Lin et al. 2020), we leverage knowledge encoded in EN by knowledge. In contrast to FedDF's strategy of ensembling all client models for global model improvement, we consider the client models from EN clients as the 'bad teacher'. The goal is to make the student model (e.g., the global model) remain distant from the 'bad teacher'. The server initializes the student model by aggregating solely the MN client models:

$$\mathbf{s}^{t+1} = \sum_{k \in \mathcal{C}^t} \frac{N_k}{N^t} \mathbf{w}_k^t, \tag{4}$$

where N_k is the number of training samples in client k and $N^t = \sum_{k \in \mathcal{C}^t} N_k$ is the total number of training samples of the selected MN clients in round t.

The initial student model is solely aggregated by MN client models, thereby preventing being affected by EN client models during model aggregation. Subsequently, the student model is updated by optimizing the negative distillation loss function $\mathcal{L}_{nd}$:

$$\frac{1}{|\mathcal{N}^t||\mathcal{D}_U|} \sum_{k \in \mathcal{N}^t, \mathbf{x} \in \mathcal{D}_U} d\left[f(\mathbf{x}; \mathbf{s}^{t+1}), g(\mathbf{x}; \mathbf{w}_k^{t+1})\right], \tag{5}$$

where $d(\mathbf{u}, \mathbf{v}) := KL\left[\sigma(\mathbf{u}), \sigma(\mathbf{v})\right]$ is the distance based on KL divergence with $\sigma(\cdot)$ the softmax activation function, $f(\mathbf{x}; \mathbf{w}) = \{f_1, f_2, ..., f_C\}$ stands for the output probability vector, and $g(\mathbf{x}; \mathbf{w}) := \{f_1^{-1}, f_2^{-1}, ..., f_C^{-1}\}$ are the reciprocals of the output. $KL(\mathbf{u}, \mathbf{v}) = (\sigma(\mathbf{u}), \sigma(\mathbf{v}))$ is the distillation loss and σ is the softmax function. This loss compels the student model's predictions to diverge from those of each EN client model. The reciprocals of the output from EN client models could include knowledge from student models, given that these models tend to produce incorrect predictions consistently. In this manner, the student model is enhanced by avoiding wrong knowledge from the 'bad teacher'. Finally, the updated student model is sent back to each client as the global model $\mathbf{w}^{t+1}$.

Algorithm 1: Federated Negative Distillation

Input: T_0 is the number of warm-up rounds; T is the total number of rounds; λ is the threshold for selecting EN clients; and K is the total number of clients.

Output: The global model $\mathbf{w}^T$ in round T

Initialize local model $\mathbf{w}_k^0$ for each client;
for $t = 1$ **to** T **do**
  // Clients execute:
  for $k = 1, ..., K$ **do**
    Update local model $\mathbf{w}_k^{t+1}$ by Eq. (7);
    Send $\mathbf{w}_k^{t+1}$ to the server;
    if $k \in \mathcal{N}^t$ *and* $t > T_0$ **then**
      Update local model $\widehat{\mathbf{w}}_k^{t+1}$ by Eq. (6);
      Send $\widehat{\mathbf{w}}_k^{t+1}$ to the server;
  // Server executes:
  Randomly select a set of active clients $\mathcal{A}^t$;
  Select $\mathcal{C}^t$ and $\mathcal{N}^t$ by Eq. (3);
  Aggregate local models to $\mathbf{s}^{t+1}$ by Eq. (4);
  if $t > T_0$ **then**
    Update $\mathbf{w}^{t+1}$ by Eq. (5);
  else
    $\mathbf{w}^{t+1} \leftarrow \mathbf{s}^{t+1}$;
  Send $\mathbf{w}^{t+1}$ and $\mathcal{N}^t$ to clients.

Client-Side Training

Once the server has identified the EN clients in round t, the subsequent client training in round $t+1$ is tailored to address their unique characteristics. Given that the data in EN clients is predominantly noisy, we employ a straightforward approach of training two distinct local models for each EN client. The first local model is trained by discarding the noisy labels entirely and updated in an unsupervised manner. This unsupervised local model is denoted as $\widehat{\mathbf{w}}_k^{t+1}$, which is updated by:

$$\widehat{\mathbf{w}}_k^{t+1} \leftarrow \mathbf{w}_k^t - \eta \nabla_{\mathbf{w}} \ell(\mathbf{w}^t; \widehat{\mathcal{D}}^k), \tag{6}$$

where $\widehat{\mathcal{D}}^k$ is the local dataset with pseudo-labels assigned by the global model $\mathbf{w}^t$ at round t. The second local model continues to train on the original local dataset $\mathcal{D}_k$:

$$\mathbf{w}_k^{t+1} \leftarrow \mathbf{w}_k^t - \eta \nabla_{\mathbf{w}} \ell(\mathbf{w}^t; \mathcal{D}^k). \tag{7}$$

These two local models are sent to the server for global model updating, with the expectation that they contribute to the global model through both model aggregation and negative distillation, respectively.

In summary, during the local training round $t+1$, every client acquires an updated supervised model $\mathbf{w}_k^{t+1}$ on its local dataset $\mathcal{D}_k$, while only the EN clients acquire the additional unsupervised models $\widehat{\mathbf{w}}_k^{t+1}$ trained on the pseudo-labeled local dataset $\widehat{\mathcal{D}}_k$. Hence, a total $k + |\mathcal{N}_k^t|$ models are uploaded to the server in each round. Given the relatively small number of EN clients, the incurred communication cost remains manageable. The ablation study confirms the efficacy of this client-side training approach.

In addition, in order to prevent an MN client that being wrongly identified as an EN client during the early training phase, we conduct a warm-up training phase in the first few rounds. During this phase, the server exclusively aggregates the MN clients without any intervention of the client training. Algorithm 1 shows the entire training process for both clients and the server.

Experiments

Experimental Settings

Datasets In the experiments, we adopt CIFAR-10 and CIFAR100 (Krizhevsky and Hinton 2009) to verify the efficacy of the proposed method. To accommodate the setting of FL with extremely noisy clients, we preprocess the training data through the following steps: (1) First, we distribute the training data across each client in a non-IID manner. Following the widely used strategy for generating non-IID clients (Yurochkin et al. 2019), we utilize Dirichlet distribution with a parameter that controls the degree of data heterogeneity. (2) Then, we assign different noise ratios to individual clients, with a subset categorized as extremely noisy clients. The noise ratio for each client is drawn from a Beta distribution $Beta(\alpha, \beta)$. (3) We add label noise to each client based on the assigned noise ratio from the Beta distribution. Due to data heterogeneity, we impose uniform noise on each client only for the classes represented within a client's local data.

For the public dataset $\mathcal{D}_U$ on the server, we use different datasets from the clients' local data. We use 128 images from CIFAR-100 as $\mathcal{D}_U$ for training CIFAR-10, and 128 images from ImageNet (Russakovsky et al. 2015) as $\mathcal{D}_U$ for training CIFAR-100. All images are randomly selected from the dataset. We simply use the official testing data split by the benchmark for global model testing.

Training Details We use ResNet-18 for CIFAR-10, and ResNet-50 for CIFAR-100 as the base model. All the compared FL methods are implemented with the same model architecture. All experiments are run by PyTorch on two NVIDIA GeForce RTX 3090 GPUs. By default, we run 100 communication rounds to present the experimental results. The total number of clients is set at 20, and an active client ratio 50% is maintained in each round. For local training, the batch size is set at 32. We use SGD with a learning rate 0.05 as the optimizer for optimization processes. The threshold λ for the identification of EN client is set at 0.12 .

Comparison with SOTA Methods

We compare the proposed FedNed with two groups of methods. (1) FL baseline methods for data heterogeneity: FedAvg (McMahan et al. 2017), FedProx (Li et al. 2020), SCAFFOLD (Karimireddy et al. 2020) and FedDyn (Acar et al. 2021); and (2) Methods for FL with label noise: FedCorr (Xu et al. 2022), RoFL (Yang et al. 2022), and FedNoRo (Wu et al. 2023a). We also evaluate the robustness of the proposed method by varying the data distribution with different noise distributions (Beta) and data heterogeneity distributions (Dirichlet). The number of EN clients drawn from

Method	**CIFAR-10**						**CIFAR-100**					
Beta	(0.1, 0.1)		(0.1, 0.3)		(0.3, 0.5)		(0.1, 0.1)		(0.1, 0.3)		(0.3, 0.5)	
Dirichlet	0.7	10	0.7	10	0.7	10	0.7	10	0.7	10	0.7	10
FedAvg	61.51	69.26	75.43	77.86	69.79	71.34	36.81	39.38	38.57	39.91	36.86	38.77
FedProx	69.81	74.69	77.72	80.31	71.77	75.87	40.10	42.81	41.04	42.39	39.52	42.32
SCAFFOLD	64.07	68.42	75.96	76.83	70.21	73.41	39.64	41.19	40.14	40.72	40.11	41.45
FedDyn	66.04	69.41	76.41	80.24	72.57	76.47	28.81	30.04	28.98	31.54	28.70	31.33
RoFL	71.64	79.05	77.03	81.26	77.58	79.04	43.28	46.07	49.36	49.41	45.95	46.42
FedCorr	74.10	78.35	81.91	<u>85.10</u>	74.55	<u>80.06</u>	40.24	44.33	<u>52.76</u>	<u>57.49</u>	<u>47.03</u>	**51.23**
FedNoRo	<u>80.25</u>	<u>80.63</u>	<u>82.41</u>	84.11	<u>77.67</u>	77.83	<u>46.33</u>	<u>47.15</u>	48.41	48.69	47.02	47.56
FedNed (Ours)	**82.83**	**85.12**	**84.97**	**86.84**	**79.43**	**82.64**	**47.85**	**48.32**	**53.74**	**57.93**	**48.21**	<u>49.68</u>

Table 1: Numerical comparison between the proposed FedNed and other FL methods with extremely noisy clients. The best results are highlighted in bold, while the second-best results are underlined.

three Beta distributions (0.1, 0.1), (0.1, 0.3), and (0.3, 0.5) are about 5-6, 2-4, and 1-2, respectively, among a total number of 20 clients, where the Beta distribution (0.1, 0.3) has the overall minimum noise ratio.

Table 1 shows the comparative results. Evidently, label noise-oriented FL methods (e.g. RoFL, FedCorr, FedNoRo, and FedNed) consistently yield superior results compared to the FL baselines (e.g. FedAvg, FedProx, SCAFFOLD, and FedDyn), as the latter solely address the challenge of data heterogeneity. By comparing FedNed with methods for FL with label noise, FedNed notably outperforms the second-best method by approximately 2%-3% on CIFAR-10, and generally exhibits better performance on CIFAR-100. In addition, it can be observed that FedNed is less sensitive to the degree of data heterogeneity. FedNed also demonstrates promising performance with different noise distributions.

Model Validation

We further delve into specific aspects related to how FedNed addresses the extreme noise challenge in FL.

Ablation Study We conduct an ablation study to assess the impact of three essential components in FedNed: the identification of extremely noisy clients (Id.), negative distillation (ND), and local pseudo-labeling (LPL). The ablation study is carried out on CIFAR-10 with twenty clients including five EN clients (with noise ratio of 99%). The baseline without any component reverts to FedAvg which simply aggregates client models on the server. With the inclusion of the identification of extremely noisy clients, we aggregate solely the identified MN clients for the global model. When negative distillation is employed, the selected EN clients are utilized by optimizing the global model via Equation (5). Regarding client-side training, if local pseudo-labeling is not adopted, we directly update the local model on its local dataset, regardless of the identification result on the server. Conversely, in cases where local pseudo-labeling is employed, we update and upload two local models for each identified EN client (c.f. Section Client-Side Training).

Table 2 shows the result of the ablation study. The most significant improvement is observed when identifying extremely noisy clients, supporting our conclusion that excluding them produces better outcomes than aggregating. Negative distillation results in an additional performance boost of approximately 2%, while local pseudo-labeling contributes around 1.5%. This validates the effectiveness of FedNed in handling extremely noisy clients. The highest performance is achieved when all components are used together.

Id.	**ND**	**LPL**	**Acc.**
✗	✗	✗	74.26
✓	✗	✗	79.95
✓	✓	✗	81.91
✓	✗	✓	81.30
✓	✓	✓	82.07

Table 2: Ablation study of major components in FedNed.

Effectiveness of EN Client Identification via Uncertainty One key factor that leads to the success of FedNed is the accuracy of EN client identification. In this regard, we evaluate the effectiveness of employing model prediction uncertainty with MC dropout as a distinguishing measure in FedNed to differentiate between MN and EN clients. Figure 3 shows the histogram of model prediction uncertainty calculated by Equation (1) for both MN and EN clients. It can be observed that the uncertainty values significantly vary between MN and EN clients, allowing for the establishment of the threshold λ within the range of $(0.12, 0.14)$ for easy segregation.

Effectiveness of Negative Distillation We have shown the accuracy improvement of utilizing negative distillation to further improve the global model by incorporating the knowledge of EN client models. In this context, we delve deeper into analyzing the nature of this improvement. Figure 4 shows the t-SNE comparison between features of a global model before and after the employment of negative distillation trained on CIFAR-10. This affirms that the enhancement induced by negative distillation stems from an improved feature representation of the global model, aiding in more coherent grouping of similar types of features.

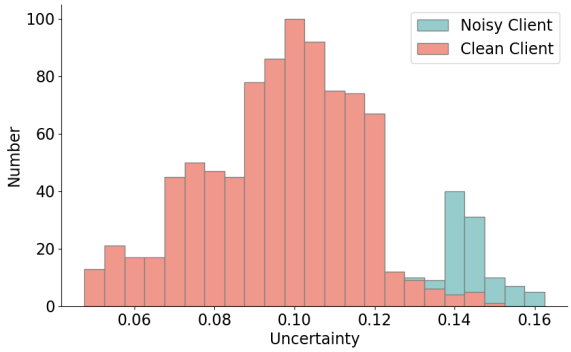
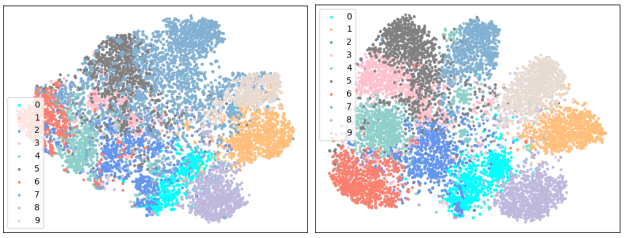


Figure 3: Histogram of model prediction uncertainty for both MN and EN clients, where the uncertainty is accumulated over all training rounds.



(a) (b)

Figure 4: Comparison in the feature spaces plotted with t-SNE. (a) FedNed without negative distillation on the server, (b) FedNed with negative distillation on the server.

Influence of the Number of Extremely Noisy Clients In this study, we have made an implicit assumption that the number of EN clients should be kept limited. The rationale behind this assumption is that excessive EN clients could lead to a notable increase in the overall noise ratio across all clients, which could render the attainment of a robust global model unfeasible. Therefore, in all previous experiments, we only set a few EN clients. Nonetheless, an intriguing curiosity led us to investigate the performance of FedNed when the number of EN clients increases substantially, possibly even constituting up to half of the client population. In this exploration, we set the total number of clients to twenty and vary the number of EN clients within the range of $[1, 3, 5, 7, 9]$. Figure 5 shows the performance degradation in accuracy as the number of EN clients increases, compared with all the other FL methods for label noise. An unexpected observation emerges: despite the continuous increase in the number of extremely noisy (EN) clients, the accuracy of FedNed remains stable, while the accuracy of other methods experiences a significant decline. It notably outperforms the baselines under these challenging circumstances. This characteristic significantly positions FedNed as a potent solution to handling extreme scenarios where many noisy clients exist.

Public Dataset Selection Considerations for further exploration include investigating the role of public datasets in FedNed. We believe that the effectiveness of the method

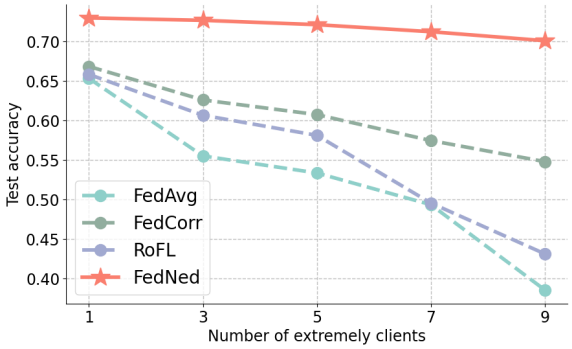


Figure 5: Comparison of performance among methods as the number of extreme noise clients increases.

Public dataset	CIFAR-10
CIFAR-100	84.97 (**2.56** ↑)
MNIST	84.89 (**2.48** ↑)
Synthetic data	83.60 (**1.19** ↑)

Table 3: The summary of the results with various public datasets shows the degree to which FedNed outperforms FedNoRo, as indicated in parentheses.

is not dependent on the choice of public datasets. We've tried i) MNIST and ii) synthetic data generated by BigGAN adopted in FedDF. The results are slightly lower than when CIFAR-100 was used as the public dataset but still higher than the second-best method, FedNoRo. Table 3 shows the results obtained using different public datasets, showcasing the adaptability of our proposed method.

Concluding Remarks

Conclusion This paper addresses the critical challenge of handling noisy labels in federated learning (FL), especially in scenarios with highly contaminated clients experiencing extreme label noise. The proposed solution, FedNed, distinguishes extremely noisy clients and incorporates them into a knowledge distillation framework, optimizing their contributions. The negative distillation process, coupled with identification by MC dropout and local pseudo-labeling, enhances the trustworthiness of the global model from noisy clients while engaging all clients for aggregation. FedNed not only outperforms existing baselines but also establishes a new state-of-the-art in FL across diverse settings.

Limitations Although the proposed FedNed mitigates performance degradation induced by extremely noisy clients in FL, a potential limitation lies in the degeneration of FedNed to FedAvg when no extremely noisy clients exist. One possible strategy involves treating FedNed as a plug-and-play module to identify extremely noisy clients, integrating with methods designed for handling mild label noise.

Acknowledgments

This study was supported in part by the National Natural Science Foundation of China under Grants 62376233, U21A20514, 62006202 and 62376235; in part by the FuXiaQuan National Independent Innovation Demonstration Zone Collaborative Innovation Platform under Grant 3502ZCQXT2022008; in part by NSFC / Research Grants Council (RGC) Joint Research Scheme under Grant N_HKBU214/21; and in part by the General Research Fund of RGC under Grants 12201321 and 12202622; in part by Guangdong Basic and Applied Basic Research Foundation No. 2022A1515011652.

References

Acar, D. A. E.; Zhao, Y.; Navarro, R. M.; Mattina, M.; Whatmough, P. N.; and Saligrama, V. 2021. Federated Learning Based on Dynamic Regularization. In *ICLR*.

Chen, Y.; Yang, X.; Qin, X.; Yu, H.; Chen, B.; and Shen, Z. 2020. FOCUS: Dealing with label quality disparity in federated learning. *arXiv preprint arXiv:2001.11359*.

Dai, Y.; Chen, Z.; Li, J.; Heinecke, S.; Sun, L.; and Xu, R. 2023. Tackling data heterogeneity in federated learning with class prototypes. In *AAAI*.

Gal, Y.; and Ghahramani, Z. 2016. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *ICML*.

Ghosh, A.; Kumar, H.; and Sastry, P. S. 2017. Robust loss functions under label noise for deep neural networks. In *AAAI*.

Han, B.; Yao, J.; Niu, G.; Zhou, M.; Tsang, I.; Zhang, Y.; and Sugiyama, M. 2018a. Masking: A new perspective of noisy supervision. In *NeurIPS*.

Han, B.; Yao, Q.; Yu, X.; Niu, G.; Xu, M.; Hu, W.; Tsang, I.; and Sugiyama, M. 2018b. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *NeurIPS*.

Jiang, H.; Kim, B.; Guan, M.; and Gupta, M. 2018. To trust or not to trust a classifier. In *NeurIPS*.

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

Karim, N.; Rizve, M. N.; Rahnavard, N.; Mian, A.; and Shah, M. 2022. Unicon: Combating label noise through uniform selection and contrastive learning. In *CVPR*.

Karimireddy, S. P.; Kale, S.; Mohri, M.; Reddi, S.; Stich, S.; and Suresh, A. T. 2020. Scaffold: Stochastic controlled averaging for federated learning. In *ICML*.

Kim, S.; Shin, W.; Jang, S.; Song, H.; and Yun, S.-Y. 2022. FedRN: Exploiting k-Reliable neighbors towards robust federated learning. In *CIKM*.

Kim, Y.; Yim, J.; Yun, J.; and Kim, J. 2019. Nlnl: Negative learning for noisy labels. In *ICCV*.

Krizhevsky, A.; and Hinton, G. 2009. Learning multiple layers of features from tiny images. In *Technical Report*, 32–33. University of Toronto.

Li, J.; Socher, R.; and Hoi, S. C. 2019. DivideMix: Learning with noisy labels as semi-supervised learning. In *ICLR*.

Li, T.; Sahu, A. K.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; and Smith, V. 2020. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2: 429–450.

Liang, S.; Huang, J.; Zeng, D.; Hong, J.; Zhou, J.; and Xu, Z. 2023. FedNoisy: Federated noisy label learning benchmark. *arXiv preprint arXiv:2306.11650*.

Lin, T.; Kong, L.; Stich, S. U.; and Jaggi, M. 2020. Ensemble distillation for robust model fusion in federated learning. In *NeurIPS*.

Lukasik, M.; Bhojanapalli, S.; Menon, A.; and Kumar, S. 2020. Does label smoothing mitigate label noise? In *ICML*.

Ma, X.; Zhu, J.; Lin, Z.; Chen, S.; and Qin, Y. 2022. A state-of-the-art survey on solving non-IID data in federated learning. *Future Generation Computer Systems*, 135: 244–258.

McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; and y Arcas, B. A. 2017. Communication-efficient learning of deep networks from decentralized data. In *AISTATS*.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; and Bernstein, M. 2015. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3): 211–252.

Shu, J.; Xie, Q.; Yi, L.; Zhao, Q.; Zhou, S.; Xu, Z.; and Meng, D. 2019. Meta-weight-net: Learning an explicit mapping for sample weighting. In *NeurIPS*.

Song, H.; Kim, M.; Park, D.; Shin, Y.; and Lee, J.-G. 2022. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*.

Tang, Z.; Zhang, Y.; Shi, S.; He, X.; Han, B.; and Chu, X. 2022. Virtual homogeneity learning: Defending against data heterogeneity in federated learning. In *ICML*.

Wang, Z.; Zhou, T.; Long, G.; Han, B.; and Jiang, J. 2022. Fednoil: A simple two-level sampling method for federated learning with noisy labels. *arXiv preprint arXiv:2205.10110*.

Wu, N.; Yu, L.; Jiang, X.; Cheng, K.-T.; and Yan, Z. 2023a. FedNoRo: Towards noise-robust federated learning by addressing class imbalance and label noise heterogeneity. In *IJCAI*.

Wu, X.; Huang, H.; Ding, Y.; Wang, H.; Wang, Y.; and Xu, Q. 2023b. FedNP: Towards non-IID federated learning via federated neural propagation. In *AAAI*.

Xia, X.; Liu, T.; Han, B.; Gong, C.; Wang, N.; Ge, Z.; and Chang, Y. 2020. Robust early-learning: Hindering the memorization of noisy labels. In *ICLR*.

Xu, J.; Chen, Z.; Quek, T. Q.; and Chong, K. F. E. 2022. Fedcorr: Multi-stage federated learning for label noise correction. In *CVPR*.

Yang, M.; Qian, H.; Wang, X.; Zhou, Y.; and Zhu, H. 2021. Client selection for federated learning with label noise. *IEEE Transactions on Vehicular Technology*, 71(2): 2193–2197.

Yang, S.; Park, H.; Byun, J.; and Kim, C. 2022. Robust federated learning with noisy labels. *IEEE Intelligent Systems*, 37(2): 35–43.

Yang, Z.; Zhang, Y.; Zheng, Y.; Tian, X.; Peng, H.; Liu, T.; and Han, B. 2023. FedFed: Feature distillation against data heterogeneity in federated learning. In *NeurIPS*.

Yao, Y.; Sun, Z.; Zhang, C.; Shen, F.; Wu, Q.; Zhang, J.; and Tang, Z. 2021. Jo-SRC: A contrastive approach for combating noisy labels. In *CVPR*.

Yurochkin, M.; Agarwal, M.; Ghosh, S.; Greenewald, K.; Hoang, N.; and Khazaeni, Y. 2019. Bayesian nonparametric federated learning of neural networks. In *ICML*.