Suppressing Noise from Built Environment Datasets to Reduce Communication Rounds for Convergence of Federated Learning

Rahul Mishra, Hari Prabhat Gupta, Tanima Dutta, and *Sajal K. Das
Dept. of CSE, IIT (BHU) Varanasi, India and *Dept. of Computer Science, Missouri S&T, Rolla, USA
{rahulmishra.rs.cse17, hariprabhat.cse, tanima.cse}@iitbhu.ac.in, and *sdas@mst.edu

ABSTRACT

Smart sensing provides an easier and convenient data-driven mechanism for monitoring and control in the built environment. Data generated in the built environment are privacy sensitive and limited. Federated learning is an emerging paradigm that provides privacy-preserving collaboration among multiple participants for model training without sharing private and limited data. The noisy labels in the datasets of the participants degrade the performance and increase the number of communication rounds for convergence of federated learning. Such large communication rounds require more time and energy to train the model. In this paper, we propose a federated learning approach to suppress the unequal distribution of the noisy labels in the dataset of each participant. The approach first estimates the noise ratio of the dataset for each participant and normalizes the noise ratio using the server dataset. The proposed approach can handle bias in the server dataset and minimizes its impact on the participants' dataset. Next, we calculate the optimal weighted contributions of the participants using the normalized noise ratio and influence of each participant. We further derive the expression to estimate the number of communication rounds required for the convergence of the proposed approach. Finally, experimental results demonstrate the effectiveness of the proposed approach over existing techniques in terms of the communication rounds and achieved performance in the built environment.

CCS CONCEPTS
• Computing methodologies → Machine learning; • Human-centered computing → Ubiquitous and mobile computing.

1 INTRODUCTION

Built environment is a human-made environment that provides space to live, work, and recreate [7]. Integrating the built environment with smart sensing generates datasets that can be exploited for effective monitoring and control. Such datasets are privacy sensitive as they possess extensive characterization of the built environment. They are also limited due to privacy concerns and a lack of smart sensing infrastructure in the built environment.

Federated Learning (FL) collaboratively trains a shared model on the server using multiple participants while keeping local and limited data decentralized to preserve privacy [17]. Prior studies utilized FL in various built environment applications, including the (electrical) power load forecasting [8] and anomaly detection [19]. Noisy labels in the datasets of the participants are one of the key challenges that decelerate the broader acceptability of FL. They degrade the performance of the trained model [28] and increase the communication rounds between the participants and the server for convergence of the FL [22]. The noisy labels also deteriorate the quality of the Weight Parameter Matrices (WPM) of the participants. Additionally, handling the unequal distribution of noisy labels among different classes becomes extremely challenging.

There exists literature on handling noisy labels in participants’ dataset to optimize the weighted contribution during aggregation [4, 12, 26]. However, training in the presence of noisy label instances in the dataset impacts the performance of the model. Prior studies removed the noisy label instances from the participants’ dataset [9, 22, 28] to minimize the impact of noise in FL. The mechanism of Bayesian checks to classify the labels as noisy or noise-free is proposed in [9], where only a few noise-free instances were used in the model training on each participant. Such few instances increase the communication rounds for convergence and degrade the performance. Exchanging the centroids between the participants and the server [22] to remove the noise reveals the characteristics of the dataset, which results in privacy compromise. Determining the noisy labels using a single model trained by a third party [28] reduces the significance of the personalization. Further, there exist methods [3, 10, 29] in centralized training to handle noisy labels; such methods do not perform well in FL due to the customary smaller sizes of the participant datasets and privacy concerns.

The number of communication rounds and the duration of each round determine the convergence time of FL. The duration of each round depends on the local epochs and the number of participants. Therefore, the number of communication rounds help estimate the required number of local epochs and participants for achieving the desired accuracy of FL in fixed time duration. The authors in [14, 16, 27] estimated the number of communication rounds for the convergence of FL, assuming all the samples have a similar impact on the training. The assumption of a similar impact may not be valid in the presence of noisy labels. This is because the random selection of samples also has noisy labels, deteriorating the performance of the trained model. For example, FedProx [13] and FedNova [23] exhibit communication rounds and accuracy conflict, where faster convergence (i.e., fewer communication rounds) come at the expense of accuracy. While existing approaches [13, 14, 16, 23, 27] estimate the communication rounds for convergence of FL with noise-free labels in the participants’ local dataset, estimating the communication rounds (i.e., the convergence time) in the presence of noisy labels with a given accuracy threshold is an important challenging problem in FL.

This paper proposes Fed-NL, a novel Federated learning approach to minimize the impact of unequal distribution of Noisy Labels in the participants dataset in built environments. Fed-NL estimates the noise ratio of the dataset for each participant and normalizes the unequal distribution of noisy labels, assuming the server has a noise-free dataset, as in existing works [4, 20, 22, 28]. Each participant demands class-wise data samples from the server.
without revealing the characteristics of the local dataset. Fed-NL also estimates the communication rounds required for the convergence of FL in the presence of noisy labels. Specifically, we investigate the following problem: *how to minimize the impact of unequal distribution of noisy labels in the dataset of each participant while estimating the communication rounds for the convergence of FL?*

**Contributions and novelty:** The main contributions and novelty of this work are as follows:

1) The proposed approach Fed-NL estimates the noise ratio in the dataset of each participant without using any prior information on noise concentration. Each participant randomly splits its dataset into multiple disjoint sub-datasets; a model is trained on one of the sub-datasets and tested on the remaining ones. Such training and testing operations are repeated on all sub-datasets to obtain a dataset with noise-free labels, thus estimating the noise ratio. This estimation is independent of server, third parties, or other participants in FL.

2) Fed-NL normalizes the estimated noise ratio to eliminate the unequal distribution of noisy labels. The class having the least noise ratio is a threshold for the remaining classes, which demand noise-free data samples from the server to achieve the threshold. The normalization reduces the noisy labels on each participant without revealing the characteristics of the local dataset.

3) Fed-NL determines the least number of samples the server transfers to a selected participant against a given class. Afterward, the participant fetches class-wise data samples from the server with the least sample count as the threshold. It neutralizes the impact of bias in the server dataset on the participant dataset.

4) Fed-NL estimates the optimal weighted contribution of each participant by incorporating the normalized noise ratio and influence of participants [26]. The optimal contribution provides robustness against the noise during aggregation. Expressions are derived to analyze the convergence of Fed-NL and the communication rounds using the following function:

\[ R = f(E, N, q_o, \epsilon_i, \mathbb{E}[\nabla L_i(\cdot, \cdot)]) \]  

where \( E, N, q_o, \) and \( \epsilon_i \) respectively denote the number of local epochs, the number of participants, the precision threshold, and the contribution of each participant \( i \in \{1 \leq i \leq N\} \). Here \( \mathbb{E}[\nabla L_i(\cdot, \cdot)] \) is the expected value of loss function gradient using randomly chosen data samples in the presence of noisy labels. (For details on the expression for communication rounds \( R \), see Theorem 5.1.)

5) The performance of Fed-NL approach is evaluated on our collected and existing datasets in transportation [21] and human activity recognition [1]. Experimental results show that Fed-NL provides robustness against noise and successfully estimates the communication rounds in the built environments. Fed-NL is also evaluated on existing image datasets [2, 11] used in federated learning.

The paper is organized as follows. Section 2 reviews the related work and Section 3 summarizes preliminary concepts. Sections 4 and 5 respectively describe the Fed-NL approach for handling noisy labels in the datasets, and estimates the communication rounds required for convergence of Fed-NL. Section 6 evaluates the performance of Fed-NL and Section 7 concludes the paper.

## 2 RELATED WORK

This section discusses the existing works on how to handle noisy labels in the participant datasets and estimate the communication rounds for convergence of FL.

### 2.1 Handling noisy labels in FL

Existing literature on handling noisy labels typically uses the following mechanisms: a) optimizing weighted contributions, b) reducing noisy labels, and c) learning with noisy labels.

The weighted contribution of the participants during aggregation at the server can be optimized to mitigate the negative impact of noisy labels in FL [4, 12, 26]. In [4], the authors highlighted the issue of noisy labels in the dataset and computed mutual cross-entropy loss among the participants and the server to assign different weight contributions. However, the technique could not handle bias due to the unequal distribution of noisy labels in the dataset. In [12], the authors proposed a framework to incorporate a data quality measurement mechanism, thereby providing robust aggregation against noisy labels in the training dataset. In the proposed framework, the noisy data instances remained confined to the participants, which hampered the performance of the trained model. The influence of participants in FL is estimated in [26] to quantify the impact on the weighted contribution. The authors removed less influential participants from FL, where eliminated participants could not improve their performance.

The negative impact of noisy labels in the dataset can be reduced by identifying and removing the instances with such labels [9, 22, 28]. The authors in [22] introduced a technique to select relevant data for model training on the participants. However, exchanging the centroids between the participants and the server [22] to remove the noisy labels reveals the characteristics of the dataset, which results in privacy compromise. The authors in [28] highlighted the incorrect annotation of participants’ datasets and proposed a method for choosing consistent decision boundaries to identify noisy and noise-free samples. Determining consistent decision boundary is effective, but identifying the noisy labels using a single unified model trained on a third party reduces the significance of the personalization in FL. A mechanism of sanitary checks is introduced in [9] to classify the class labels as noisy or noise-free. The authors utilized only a few noise-free instances for training the model using the student-teacher training approach. However, removing noisy labels from the dataset and working with limited samples increases the local epochs for training and leads to performance compromise of FL [28]. Additionally, model training with the unequal noise distribution among the participants and server (i.e., students and teacher) degrade the performance of FL.

In prior studies, several methods have been designed to provide effective learning despite noisy labels in the dataset [3, 10]. In [10], a set of trusted data with clean labels is used to resolve the negative impact of noisy labels. A classifier is initially built using limited and noise-free data, followed by training on all the instances. However, considering noisy data after initial training on noise-free data hampers the performance of the trained model in FL. The classification problem with training instances having randomly corrupted labels is addressed in [15], where the authors introduced the importance of data instances re-weighting technique to best utilize
the loss functions designed for traditional classification problems. Similar to [10], the training in FL used noisy labeled data instances, which degraded the performance of the trained model.

2.2 Communication rounds in FL
The existing work have estimated the number of communication rounds required for convergence of FL [14, 16, 27]. As regards to the convergence of FL, the authors in [14] estimated the number of communication rounds in FL, considering participants’ datasets with noise-free labels. Similarly, the authors in [16] analyzed the role of three control variables in FL, i.e., the communication rounds, local epochs, and the number of participants to determine the number of communication rounds for convergence of FL. The derived expression for communication round established a reciprocating relation between communication rounds and local epochs, i.e., when local epochs increased, the number of communication round decreased. The authors in [27] extended the estimation of communication rounds in [14] for iid and non-iid datasets with partial and complete involvement of all the participants. They proved that the linear speedup is achieved under both iid and non-iid assumptions.

The estimation of communication rounds for convergence of FL in [14, 16, 27] is applicable for participants’ datasets with noise free labels, where the assumption for a similar impact of all the samples in the dataset is valid. However, such an assumption of a similar impact may not be applicable for participants’ datasets with noisy labels. Therefore, estimating communication rounds with noisy labels is tedious and requires redefining the variables in expression for communication rounds of FL with noise-free labels.

3 PRELIMINARIES
This work considers an FL scenario for multi-class classification problem with c classes of set [c] = {1, 2, · · · , c}. A set of N participants in FL is denoted by [p] = {p1, p2, · · · , pN}. Each participant pi has dataset Di with ni number of instances and classes [c], where 1 ≤ i ≤ N. Similarly, let D̃ denote the server dataset with classes [c]. Let D̃k and D̃ are the sub-datasets of D̃ for class k, respectively, ∀k ∈ [c]. This work assumes that D̃k may have noisy labels, ∀i ∈ [1 ≤ i ≤ N]. Let (xij, yij) denotes an instance of dataset Di, where 1 ≤ j ≤ ni. Let yij and y′ ij denote the true and predicted labels by the model of an instance xij, respectively, where yij ∈ [c]. An instance xij ∈ Di has noise-free label, if the observed yij matches with yij; however, yij is usually unavailable. Let xij denotes an instance of dataset Di at participant pi having class label yij, where 1 ≤ i ≤ N, 1 ≤ j ≤ ni, and yij ∈ [c]. The instance xij is said to have noisy label if either of the following conditions hold: a) yij is mislabeled as any other class of Di, i.e., yij ∈ [c] − yij, where yij denotes the true class label. b) yij is an arbitrary class label, i.e., yij /∈ [c]. This work assumes that the server comprises a dataset with noise-free labels. The existing literature in FL inspires this assumption [4, 20, 22, 28], where the server is assumed to possess a dataset. Moreover, the assumption does not reveal the characteristics of participant datasets; thus, it maintains data privacy in FL.

• Federated learning: Each participant pi in FL receives a model from the server and trains it using dataset Di. The training is performed for E number of local epochs and minimizes the loss function L̃(w̃i, D̃) of participant pi, where w̃i is the WPM of pi. L̃(w̃i, D̃) is estimated as: L̃(w̃i, D̃) = 1 ni ni−1 j=0 L̃ij(w̃ij, x̃ij), where x̃ij ∈ D̃, 1 ≤ j ≤ ni, and 1 ≤ i ≤ N. Participant pi estimates L̃(·, ·) and transfers L̃(·, ·) and w̃i to the server for global aggregation. Next, the server estimates global loss function as L̃(w, D̃) = ni=1 ñni ñni L̃ij(w̃ij, x̃ij) using weighted aggregation of the local losses and WPM of participants, where Ñ = ni + ñ + · · · ñN.

The server broadcasts the aggregated WPM to all the participants for the next round of training. The process of local training and global aggregation is orchestrated for R global iterations to achieve a trained model for all the participants. At each global iteration t ∈ R the local loss function and WPM are denoted as L̃t(·, ·) and w̃t for participant pt, respectively, where 1 ≤ i ≤ N. Similarly, the server has aggregated loss function and WPM as L̃(·, ·) and w̃ at iteration t, respectively. During local training for E epochs the WPM w̃ at t ∈ R of participant pt is updated as: w̃t,lk = w̃t,lk−1 − η̃L̃(w̃t,lk−1, D̃), ∀e ∈ E, where η̃ is the learning rate and w̃t,lk is initialized as: w̃t,ℓ0 ← w̃t−1.

• Influence of participants: The influence of pt on aggregated WPM w̃ at iteration t ∈ R, denoted as δ̃t, can be estimated by removing the WPM w̃l from aggregation, where 1 ≤ l ≤ N. Let [p′t] of N denotes the set of participants available at iteration t. Let N([p′t], pi) denotes the sum of instances of all the participants excluding pi then aggregated WPM at iteration t is given as [26]:

w̃t([p′t]|pi) = ni=1 t([p′t]|pi) ñni=1 N([p′t]|pi) w̃i.

The influence δ̃t of pt when it is permanently removed from the training is given as:

δ̃t = w̃t([p′t]|pi) − w̃t.

• Distribution of noisy labels: There are two types of distribution of noisy labels in the dataset of the participants [3], i.e., symmetric and asymmetric. The data instances in the symmetric distribution of the noisy labels having the same distance to the class boundary are equally likely to be corrupted. The noise ratio of participant pi, denoted as βi, is the fraction of the noisy instances to the total instances of pi. The noise transition matrix P for symmetric distribution of the noisy labels is defined as: Pkk = 1 − βi and Pkl = βi/(c − 1), where k, l ∈ [c]. In the asymmetric distribution, the data instances with the same class boundary distance are not equally likely to be corrupted. P for the asymmetric distribution of noisy labels is given as: ∑k,l∈[c] Pkl = 1, where Pkk > Pkl, others Pkl may be unequal, and ∀k /∈ l.

4 FED-NL: FEDERATED LEARNING APPROACH FOR HANDLING NOISY LABELS
This section presents the Federated Learning approach with Noisy Labels (Fed-NL). An overview of the Fed-NL is shown in Figure 1. Fed-NL starts with the estimation of the noise ratio on each selected participant in FL. Next, we introduce the noise ratio normalization mechanism because the obtained noise ratio of a participant may not be equally distributed among all the class labels.
We also determine the least data samples the server can transfer to each participant class-wise. Further, we estimate the optimal contribution of each participant as per the noise ratio and influence. We summarize these steps in Algorithm 1. Finally, we derive the expression to estimate the communication rounds \( R \) for convergence of Fed-NL.

### 4.1 Estimating noise ratio at each participant

Each participant in Fed-NL randomly splits its dataset into multiple disjoint sub-datasets; a model is trained on the one sub-dataset and tested on the remaining. The authors in [3] split the dataset into two sub-datasets to determine noisy labels by alternatively training and testing on the first and second sub-dataset. It provided one predicted and one existing label to identify noise-free labels. However, it would not work if the existing and the predicted labels are noisy; the label is marked as noise-free. To overcome the above limitation, the estimation of noise ratio in Fed-NL starts with the random split of \( D_k \) of participant \( p_i \) into three sub-datasets having a nearly equal instances, denoted as \( D_k^1 \), \( D_k^2 \), and \( D_k^3 \), where \( 1 \leq i \leq N \).

A participant \( p_i \) initially trains the local model \( M_i \) on \( D_k^1 \) and the data instances of \( D_k^2 \) and \( D_k^3 \) are tested using the trained \( M_i \), \( \forall i \in \{1 \leq i \leq N\} \). The testing operation results in a set of output labels for \( D_k^2 \) and \( D_k^3 \), respectively. \( p_i \) next re-initializes and trains \( M_i \) on \( D_k^2 \) and tests \( D_k^1 \) and \( D_k^3 \) instances on trained \( M_i \) to obtain set of output labels. \( p_i \) finally re-initializes and trains \( M_i \) on \( D_k^3 \) and tests \( D_k^1 \) and \( D_k^2 \) to obtain set of output labels. Now, we have three labels for each instance of any sub-dataset, i.e., one existing and two predicted. An instance of a sub-dataset is noise-free if its existing label is the same as two predicted labels, as shown in Example 4.1. Fed-NL uses this mechanism to estimate class-wise noise-free labels in each sub-dataset and assign to a set \( S_{ik}^q \) where \( 1 \leq i \leq N, k \in \{c\}, \) and \( q \in \{1, 2, 3\} \). For example, all the instances with noise-free labels in \( D_k^1 \) are assigned to \( S_{ik}^1 \). Similarly, we determine \( S_{ik}^2 \) and \( S_{ik}^3 \). Further, we obtain a noise-free set \( S_{ik} \) by taking the union of sets \( S_{ik}^1, S_{ik}^2 \) and \( S_{ik}^3 \), i.e., \( S_{ik} = S_{ik}^1 \cup S_{ik}^2 \cup S_{ik}^3 \). The removed set \( R_{ik} \) is estimated by following expression: \( R_{ik} = D_{ik} \setminus S_{ik} \).

**Example 4.1.** Let us consider a binary classification task, where a model is trained on a noisy dataset of any participant. An instance of the dataset, can have the following possibilities of labels using the three-fold split of the dataset.

| Existing | Predicted 1 | Predicted 2 |
|----------|-------------|-------------|
| Noisy    | 0           | 1           |
| Noisy    | 0           | 0           |
| Noisy    | 0           | 1           |
| Noise-free | 0         | 0           |

We set the last row of (4) as noise-free as the existing label and predicted labels (Predicted 1 and Predicted 2) from alternative training and testing are same. In other rows, there is conflict among existing and predicted labels. The conflict is due to noisy labels or improper training owing to the insufficiency of the dataset.

The noise ratio of participant \( p_i \) for class \( k \) is estimated as: \( \beta_{ik} = \frac{|R_{ik}|}{|D_{ik}|} \), where \(| | \) is the cardinality. The participant \( p_i \) estimates the noise ratio for all the classes. Similarly, all the participants estimate the class-wise noise ratio. Procedure 1 summarizes the steps involved in estimating the noise ratio in Fed-NL.

### 4.2 Class-wise normalization of noise ratio and data samples from server to participant

A class with minimum noise ratio (\( \beta_{ik} = z_i \)) does not demand data from the server, and its noise ratio acts as a threshold for other classes. The participant \( p_i \) for other classes which have noise ratio more than \( z_i \), demands a fraction \( \mathcal{F}_{ik} \leftarrow \left( \beta_{ik} - z_i \right) \) of dataset \( D_{ik} \) from the server. The server uniformly at random selects \( \mathcal{F}_{ik} \) portion of data instances from \( D_{ik} \) and sends to the participant. However, the server dataset might be biased where data instances...
We consider the test dataset of server, denoted as $D^t$, to evaluate influence $\delta_{ij}$ (given in (3)) of participant $p_i$ at global iteration $t$ ($t \in \mathcal{R}$). The exact estimation of $\delta_{ij}$ is only possible when we remove the WPM of participant $p_i$ from the entire training process. After that, the retraining of the shared model is performed using the remaining participants. However, retraining for each participant at each global iteration is prohibitively inefficient. Therefore, we use the expression for the estimated value of $\delta_{ij}$, given in [26]. Let $\delta_{ij}'$ denote the estimated value of $\delta_{ij}$, which is denoted as:

$$\delta_{ij}' = Q_{ij}^t \delta_{ij}^{-1} + w^t ((p^t_i | p_i) - w^t),$$

(5)

for some classes are low and high for others. Specifically, the server may be incompetent in fulfilling the demand of $|F_{ik}|$ data instances, for a few classes $c$, from the participant $p_i$. It results in bias of the class instances on the participant. Therefore, we identify the least possible data instances the server could transfer to the participant among all classes, denoted as $\Delta_{ik}$, as shown in lines 8 - 17 of Procedure 2. Further, the server transfers class-wise and uniformly at random selected $\Delta_{ik}$ data instances to the participant $p_i$ for classes having noise ratio $\geq z_i$.

The update provides robustness against non-convex loss functions and reduces the computation for the Hessian matrix. This work adopts a similar strategy to update the basic estimator.

This work considers datasets of the participants with noisy labels; therefore, we use effective noise-free data instances to estimate $Q_{ij}^t$. We replace $n_i$ with $n_i / \bar{\beta}_i$ and $N((p^t_i | p_i)$ with $M((p^t_i | p_i)$ to obtain $Q_{ij}^t'$, where $M((p^t_i | p_i) = (n_1 / \bar{\beta}_1 + n_2 / \bar{\beta}_2 + \ldots + n_N / \bar{\beta}_N)$ $- n_i / \bar{\beta}_i$. Using $Q_{ij}^t'$ and including noise ratio in the estimation of $w^t((p^t_i | p_i)$ and $w^t$, we can obtain modified influence of participant $p_i$, denoted as $y_i'$. Later, we use $y_i'$ to estimate the contribution $e_i$ of participant $p_i$ at global iteration $t (t \in \mathcal{R})$, $e_i$ is estimated using following expression: $e_i = 1 / y_i'^t (Q_{ij}^t)$, where $Q_{ij}^t = 1 / y_i'^t + 1 / y_i'^t + \ldots + 1 / y_i'^t$. We use $e_i$ to estimate aggregated WPM $w^t$ on the server at global iteration $t (t \in \mathcal{R})$, which is given as: $w^t = \sum_{i=1}^{N} e_i w_i'$. Procedure 3 summarizes the steps of estimating the weighted contribution of all the participants at the server.

Finally, the Fed-NL algorithm (Algorithm 1) uses Procedure 1, Procedure 2, and Procedure 3 to handle the noisy labels in the datasets of participants.

### 5 CONVERGENCE ANALYSIS AND COMMUNICATION ROUNDS OF FED-NL

This section presents the convergence analysis and derives the expression for the communication rounds required for convergence of Fed-NL. Apart from the convergence analysis of FedAvg in [14] on a noise-free dataset, the noisy labels in the participant’s dataset make a significant variance in the local loss function $L_i(\cdot)$ of each participant and aggregated global loss function $L(\cdot)$.

Let $\delta_{ij}'$ denotes the uniformly and randomly selected sample from the dataset of participant $p_i$ at global iteration $t$, where $1 \leq t \leq \mathcal{R}$.
Algorithm 1: Fed-NL Algorithm

Input: Set \( [p] \) of \( N \) participants with datasets;
Output: Trained model on \( p_i \) (1 \( \leq i \leq N \));
1. Server builds a model using random weights;
2. Server broadcasts the WPM to all the participants;
3. for each participant \( p_i \in \{ p_1, p_2, \ldots, p_N \} \) do
   4. Call Procedure 1 to estimate noise ratio;
   5. Call Procedure 2 to normalize noise ratio & data from server;
   6. Call Procedure 3 to estimate weighted contribution \( e_i^t \) for each participant \( p_i \) at \( t \);
7. while \( 1 \leq t \leq R / \text{global iterations} \) do
   8. for each participant \( p_i \in \{ p_1, p_2, \ldots, p_N \} \) do
      9. Calculate \( w_i^{t+1} \) send to the server;
      10. \( w_i^{t+1} = \sum_{j=1}^{N_j} e_i^j w_i^j \);
   11. Broadcast \( w_i^{t+1} \) to all the participants;
   12. \( t \leftarrow t + 1 \);
13. return Trained model on each participant;

Let \( \nabla L_i(\phi_i^t, w_i^t) \) and \( \nabla L_i(w_i^t) \) denote the gradients of loss function \( L_i(.) \) on \( \phi_i^t \) samples and entire samples of the dataset, respectively. Assumptions 1 and 2 for loss functions are \( L \)-smooth and \( \mu \)-strongly convex, respectively, in [14] are generalized; thus, we directly employed them in this work. Assumptions 3 and 4 in [14] are not applicable for the participant with noisy labels as \( \| \nabla L_i(\phi_i^t, w_i^t) - \nabla L_i(w_i^t) \| \) may not be same for different set of samples drawn randomly. Thus, we use expected value of gradient of loss function \( \mathbb{E}[\nabla L_i(\phi_i^t, w_i^t)] \) for samples randomly drawn from the dataset at \( t \). We redefine Assumptions 3 and 4 of [14] as follows:

1. Variance of gradients on each participant \( p_i \) for noisy labels is bounded as: \( \mathbb{E}[\mathbb{E}[\nabla L_i(\phi_i^t, w_i^t) - \nabla L_i(w_i^t)]^2] \leq \sigma_i^2 \).
2. \( \mathbb{E}[\mathbb{E}[\nabla L_i(\phi_i^t, w_i^t)]]^2 \leq G^2 \). Using property of expected value, \( \mathbb{E}[\mathbb{E}[\nabla L_i(\phi_i^t, w_i^t)]^2] \leq \mathbb{E}[\nabla L_i(\phi_i^t, w_i^t)]^2 \leq G^2 \).

Using the above assumptions, we obtain a relation between desired precision \( q_0 \), local epoch \( E \), and global iterations \( R \), as given in Theorem 5.1. The desired precision is defined as: \( q_0 = \mathbb{E}[\mathbb{E}[\nabla L_i(w_i^R)]] - \mathbb{E}[\nabla L_i(w_i^L)] \), where \( w_i^R \) is the aggregated weight at final global epoch \( R \) and \( w_i^L \) is minimum and unknown value of \( L \) at the server. Let \( L_i^* \) is the minimum value of \( L_i \) at \( p_i \), where \( \forall i \in \{ 1 \leq i \leq N \} \). This work assumes an iid dataset at the participant and non-iid datasets among different participants; thus, overall datasets in FL are non-iid. We define: \( \Gamma = \Gamma^* - \sum_{i=1}^{N} L_i^* \). \( \Gamma \) quantifies the degree of non-iid, and it goes to zero as the data instances increase.

\section{Theorem 5.1} Let \( \alpha = \max(8L/\mu, E) \) and \( T \) is SGD operations on a participant in FL then we obtain following relation of communication rounds \( R \) with desired precision \( q_0 \) and local epoch \( E \):

\[
R = \frac{1}{E} \left[ \frac{L}{2E^2 q_0} \left( 4B + \mu^2 \alpha \mathbb{E}[\|w_1 - w^*\|^2] + 1 - \alpha \right) \right],
\]

where \( B = \sum_{i=1}^{N} e_i^2 \left( \mathbb{E}[\mathbb{E}[\nabla L_i(\phi_i^t, w_i^t)] - \nabla L_i(w_i^t)]^2 \right) \right) + 6L\Gamma + 8(E - 1)^2 \left( \mathbb{E}[\mathbb{E}[\nabla L_i(\phi_i^t, w_i^t)]^2 \right).
\]

The proof of Theorem 5.1 is given in the following section.
Using (14) we get the desired precision \( q_0 \) after \( R \) rounds as:
\[
q_0 = \mathbb{E}[L(w^R)] - L^* \leq \frac{L/2\mu^2}{\alpha + T - 1} \left( 4B + \mu^2 \alpha \mathbb{E}[\|w_1 - w^*\|^2] \right). \tag{15}
\]

Similar as the convergence of FedAvg in [14], (15) indicates Fed-NL converges to a global optimum at the rate of \( O(1/T) \). Using the upper bound of \( q_0 \) and \( T = R\mathcal{E} \), we obtain:
\[
q_0 = \frac{L/2\mu^2}{\alpha + R\mathcal{E} - 1} \left( 4B + \mu^2 \alpha \mathbb{E}[\|w_1 - w^*\|^2] \right). \tag{16}
\]

Upon solving (16), we obtain communication round \( (\mathcal{R}) \) as follows:
\[
\mathcal{R} = \frac{1}{E} \left[ \frac{L}{2\mu^2 q_0} \left( 4B + \mu^2 \alpha \mathbb{E}[\|w_1 - w^*\|^2] \right) + 1 - \alpha \right]. \tag{17}
\]

Hence proved. □

6 PERFORMANCE EVALUATION

This section first describes the collected and existing datasets (SHL [21], HAR [1], FEMNIST [2], and CIFAR-10 [11]) and the baseline techniques for ablation studies. Next, we discuss implementation details and validation metrics. Finally, we conduct the experimental evaluation to validate the performance of the Fed-NL.

6.1 Datasets and baseline techniques

We collected a Locomotion Mode Recognition (LMR) dataset to recognize six modes, i.e., bicycle, bike, car, auto-rickshaw, bus, and train. We developed an android application to facilitate data collection that uses three onboard sensors, including accelerometer, gyroscope, and magnetometer. We appointed 40 volunteers during LMR data collection. The data instances were recorded for 60 seconds, which is enough time to provide sufficient distinguishable characteristics among different locomotion modes. Volunteers were instructed to collect 200 data instances per locomotion mode during data collection. Thus, we obtain 48000 data instances in the LMR dataset, where 40 volunteers x 6 locomotion modes x 200 repetitions = 48000. To automatically annotate (or tag) the data instances, the volunteers were instructed to capture an image of the locomotion mode before recording the data. The volunteers also tag the data instances manually to check the correctness of the automatic annotation. Using manual tagging, we observed that the automated annotation introduced noisy labels distributed asymmetrically, generating the need to handle such labels. Next, we considered the existing SHL [21] dataset, which used similar sensors as LMR (i.e., accelerometer, gyroscope, and magnetometer) and possessed instances for more classes than LMR. Afterward, we considered HAR [1] dataset due to its broader applicability in FL literature during experiments. HAR also utilized a similar set of sensors (accelerometer, gyroscope, and magnetometer) as the LMR dataset for human activities recognition. Finally, we considered widely used image-based FEMNIST [2] and CIFAR-10 [11] datasets in FL. These datasets were selected due to free accessibility, real-life acquisition, and correct annotations. FEMNIST is built by dividing the existing Extended MNIST [5] depending upon the writer of the digit/character. CIFAR-10 dataset [11] comprises 60000 images of 10 different classes. Each class has 6000 images.

Since the collected and existing datasets have noise-free labels; thus, we explicitly add asymmetric noisy labels in the dataset to test Fed-NL in the real world. To inject noisy asymmetric labels into the dataset, we follow the label transition to design some of the structure of real mistakes for similar classes, as discussed in [3].

Section 2 illustrated that most of the existing literature on FL to handle noisy labels are considered optimizing weighted contribution [22] and reducing noisy label instances [4, 28] techniques. We selected techniques [4, 17, 22, 28] as baselines for ablation studies, denoted as Base1 [22], Base2 [28], Base3 [4] and Base4 [17], to compare and evaluate the effectiveness of Fed-NL.

6.2 Implementation details

We used the DeepZero model given in [18] during the experimental evaluation on LMR and SHL datasets. We used Convolutional Neural Networks (CNN) as the classifier for evaluating Fed-NL on HAR, FEMNIST, and CIFAR-10 datasets. CNN model is similar to the one used in [25]. We selected 40 devices as participants in Fed-NL. We combined and randomly shuffled the dataset collected from 40 volunteers and split the shuffled data into 40 parts to obtain non-iid datasets for \( N = 40 \) participants. Similarly, we divide the existing datasets (SHL, HAR, FEMNIST, and CIFAR-10) into 40 parts for all the participants. We set \( L^* = 0.7999, E = 20, N = 40, \) and used other parameters similar as given in [14] to solve (17). We store noise-free collected and existing datasets on the server with all instances; thus, the mismatch between the distribution of participants and the server’s dataset did not persist. The considered participating devices are smartphones with similar RAM and processing capacity. They are operating over the same Wi-Fi connection. We used a Dell server with an Intel Dual Xeon processor operating over a Gigabit Ethernet connection as server.

6.3 Validation metrics

This work used standard metrics to evaluate the performance of Fed-NL: F1-score and accuracy. We also derive the following metrics:

- **Local performance metric:** This metric determines the performance of participant \( p_i \) on its dataset \( D_i \), i.e., F1-score, accuracy and confusion metric on testing sub-dataset of \( D_i \). It answers the following question: how much improvement or deterioration in the performance of participants are observed by using the Fed-NL?

- **Global performance metric:** It estimates the performance of participants on server dataset \( D \). It answers: how well the trained model on the participant performs on the server’s dataset?

6.4 Experimental results

6.4.1 Impact of noise ratio. This experiment aims to determine the impact of noise ratio on the local and global performance of a participant in FL. During the experiment, we considered the proposed Fed-NL and the baseline techniques (Base1, Base2, Base3, Base4). We used the asymmetric noisy labels on the collected LMR dataset.

Table 1(a) illustrates the local performance of a participant at different noise ratios using the collected LMR dataset and the communication rounds = 120. It shows that Fed-NL gives higher accuracy than Base1, Base2, Base3, and Base4 for all considered ratios. Base1 removed noisy samples from the dataset, reducing the training data size. Base2 and Base3 did not eliminate the data instances with noisy labels from their dataset; thus, the model learned with
We observed that the standard deviation in F1-score and accuracy is around 83%, with the noise ratio. Because with the increase in the noise ratio, the false negatives and false positives are crucial and give less uniform distribution of noisy labels and vice-versa for class a because the training instances of class a are less hampered during normalization to exploit the correctly annotated data instances of the server’s dataset to improve the performance.

Observation: The first observation from the result is that the noise ratio deteriorates local and global performance. Fed-NL achieves high performance by removing instances with noisy labels and adding correctly annotated instances from the server. Next, we observed that testing on personalized data instances achieves higher performance than testing on generalized data instances. Finally, F1-score is more effective in noisy environments than accuracy.

### 6.4.2 Impact of noise ratio normalization

This experiment aims to highlight the requirement of noise ratio normalization. We considered the collected LMR dataset, asymmetric noisy labels, noise ratio 0.2, communication rounds R = 120 using (17), and local F1-score metric during the experiment. We used notations a1, a2, a3, a4, a5, and a6 to denote class labels bicycle, bike, car, auto-rickshaw, bus, and train of LMR dataset, respectively.

Figure 2(a) illustrates the confusion matrix of the collected LMR dataset, where the noise ratio is un-normalized. The local F1-score of class a3 (“car”) is highest and class a5 (“bus”) is lowest. It is because the training instances of class a3 are less hampered with asymmetric distribution of noisy labels and vice-versa for class a5. The overall local F1-score without normalization is around 83%, less than the normalization case of around 89% at a noise ratio of 0.2. Figure 2(b) depicts the confusion matrix with normalized noise ratio on the participant’s dataset. The result illustrates that the normalization of the noise ratio improves the F1-score of all the classes in the dataset. The improvement in the F1-score of class a3 is minimum (≈ 1.0%) as its instances are less hampered during noise insertion. However, the improvement on the F1-score of a3 is highest around 10%. During normalization, the participant gets larger correctly annotated instances from the server dataset for a3.

Observation: Noise ratio normalization is crucial in improving class-wise performance. It ensures the uniform distribution of noisy labels among different classes in the dataset. Fed-NL uses noise ratio normalization to exploit the correctly annotated data instances of the server’s dataset to improve the performance.

### 6.4.3 Impact of dataset

This experiment aims to determine the impact of LMR, SHL, HAR, FEMNIST, and CIFAR-10 datasets on performance. We considered the asymmetric noisy labels in datasets and rounds R as (Dataset → R): LMR → 120, SHL → 120, HAR → 100, FEMNIST→ 100, and CIFAR-10→ 150.

Table 2(a) illustrates local F1-score decreases with the noise ratio for all the datasets. The reason behind this decrement is mentioned in Section 6.4.1. The F1-score of the SHL dataset decreases much more rapidly than that of the collected LMR. It is because the class labels in SHL possess imbalanced class labels that result in the non-uniform distribution of noisy labels, and performance is compromised. At a low noise ratio, e.g., 0.1, the local F1-score on LMR

| Approach/Technique | Noise ratio (β) | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 |
|---------------------|----------------|-----|-----|-----|-----|-----|-----|
|                     | F1             |     |     |     |     |     |     |
| Fed-NL              | 81.9           | 63.5 | 2.31 | 2.17 | 1.7 | 0.85 |
|                     | 7.34           | 8.04 | 4.19 | 2.23 | 2.79 | 1.76 |
|                     | 3.29           | 5.79 | 6.92 | 4.55 | 2.19 |
|                     | 1.79           | 4.28 | 3.27 | 84.5 | 2.94 | 3.1 |
|                     | 2.17           | 1.83 | 1.09 | 2.83 | 30.39 | 49.48 |
|                     | 1.51           | 1.35 | 0.94 | 1.48 | 3.37 | 5.24 |

(a) Un-normalized noise ratio.

| Approach/Technique | Noise ratio (β) | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 |
|---------------------|----------------|-----|-----|-----|-----|-----|-----|
|                     | F1             |     |     |     |     |     |     |
| Fed-NL              | 91.7           | 3.04 | 1.07 | 0.92 | 1.77 | 1.87 |
|                     | 2.93           | 89.7 | 4.11 | 3.17 | 1.44 | 2.86 |
|                     | 2.01           | 29.0 | 5.6 | 2.73 | 3.71 | 5.29 |
|                     | 1.31           | 1.91 | 2.12 | 89.4 | 1.21 | 2.47 |
|                     | 0.92           | 0.86 | 1.33 | 2.71 | 0.71 | 5.21 |
|                     | 1.13           | 1.57 | 1.27 | 1.07 | 2.77 | 86.3 |

(b) Normalized noise ratio.
and SHL datasets are comparable, as shown in Table 2(a). However, at a high noise ratio, e.g., 0.6, the local F1-score on LMR is significantly higher than the SHL dataset. The impact of unbalanced class labels at a low noise ratio is less, which increases with the increase in the ratio. HAR, FEMNIST, and CIFAR-10 datasets have a large number of instances; thus, the model on these datasets achieves a higher F1-score, in contrast with LMR and SHL datasets. Moreover, Fed-NL achieved F1-score > 80% even at the noise ratio of 0.6 on these datasets. It is because the noisy labels are successfully handled by demanding data instances from the server. Similarly, Table 2(b) illustrates the impact of different datasets on global F1-score.

Observation: An interesting observation from the result is that the unbalanced classes in the dataset with noisy labels result in higher performance compromise during training. It also indicates that if data instances of participants are noisy with unbalanced class labels, achieving higher performance is tedious. Therefore, an unbalanced dataset at the participant demands a significant amount of server data instances during normalization. Further, when the number of training instances in the dataset is high, the achieved F1-score is high.

Table 2: Illustration of impact of different datasets, collected LMR, SHL, HAR, FEMNIST, CIFAR-10, on local and global F1-scores.

| Dataset | Local F1-score | Global F1-score |
|---------|----------------|----------------|
| Noise ratio | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 |
| LMR | 92.23 | 88.53 | 81.43 | 74.76 | 65.43 | 52.23 |
| SHL | 91.31 | 87.61 | 78.33 | 69.87 | 61.43 | 48.07 |
| HAR | 94.20 | 92.64 | 90.61 | 88.93 | 86.29 | 84.11 |
| FEMNIST | 95.10 | 93.27 | 91.03 | 89.20 | 87.19 | 86.37 |
| CIFAR-10 | 92.71 | 90.39 | 89.71 | 87.76 | 85.73 | 82.91 |

6.4.4 Impact of weighted contribution. The objective of this experiment is to illustrate the effectiveness of the Fed-NL in estimating the weighted contribution of the noisy participants. In doing so, we considered three schemes S1 (Base4 [FedAvg [17]]), S2 (Base3 [4]), and S3 (Fed-NL). In this work, we assigned 2400 instances of collected LMR and 1631 instances of SHL on all 40 participants’ devices for local training. Noisy labels are explicitly inserted in the dataset of the participant. We also assumed 20 out of 40 participants have noisy labels, i.e., 50% participants are noisy. We considered contribution ratio calculated as: contribution on x noise ratio in dataset/contribution on noise-free dataset, where x = {0.1, 0.2, · · · , 0.6}.

Figure 3(a1) illustrates the weighted contribution of a participant at different noise ratios on the considered schemes, i.e., S1, S2, and S3 using LMR dataset. It shows that the weighted contribution of Scheme S1 is the same for any noise ratio. It is because the weighted contribution of S1 depends only on the number of local data instances. However, S2 and S3 schemes reduce the contribution with the increase in noise ratio. S2 used cross loss among server and participant to determine weighted contribution, whereas S3 (proposed) used noise ratio and participant influence to determine contribution. Figure 3(a2) depicts the reduction in local F1-score due to the increase in noise ratio. The result illustrates that S3 and S1 achieved the highest and lowest performance, respectively. It is because the S1 did not incorporate any mechanism to handle noisy labels, whereas S3 used noise ratio normalization and effectively determined contribution. Figures 3(b1) and 3(b2) illustrate the contribution and local F1-score on SHL dataset, respectively.

Observation: An interesting observation from the result is that estimating the optimal weighted contribution at different noise ratios is crucial to achieving significant performance. It also indicates that considering the influence of participant and noise ratio while assessing weighted contribution helped in improving the performance.

Figure 4(a) illustrates the impact of the communication round on the local accuracy of the participant using the collected LMR dataset. Noisy labels are explicitly inserted in the dataset of the participant. We also assumed 20 out of 40 participants have noisy labels, i.e., 50% participants are noisy. We considered contribution ratio calculated as: contribution on x noise ratio in dataset/contribution on noise-free dataset, where x = {0.1, 0.2, · · · , 0.6}.

Figure 3: Impact of weighted contribution on the datasets.

6.4.5 Number of communication rounds. The objective of this experiment is to estimate the number of communication rounds (or global iterations) required for convergence of Fed-NL, Base1 [22], Base2 [28], Base3 [4], and Base4 (FedAvg [17]). We demonstrate the improvement in local accuracy by increasing the communication rounds on the collected LMR and SHL datasets at the noise ratio of 0.1. We further extend the experiment to study the impact of noise ratio on communication rounds using the LMR dataset.

Figure 4(a) illustrates the impact of the communication round on the local accuracy of the participant using the collected LMR dataset. We observed a rapid increment in the local accuracy of the Fed-NL up to 43 epochs and a marginal increment afterward. Later, we observed that after 60 epochs, the performance improvement is negligible. It indicates that the Fed-NL converged after 60 epochs and achieved an accuracy of 91.43% at the noise ratio of 0.1. Similarly, Base1 converged after 80 epochs and achieved an accuracy of 84.21%. Base2 and Base3 acquired the local accuracy of 89.22% and 86.51%, respectively, upon convergence. Base4 (FedAvg) required the highest number of communication rounds and achieved the least accuracy due to the absence of the noise handling mechanism.
The higher performance and least epochs for convergence of Fed-NL are due to effective estimation of weighted contribution and noise ratio normalization via server dataset. Similar observation can be made for the SHL dataset, as shown in Figure 4(b).

![Figure 4: Communication rounds on Fed-NL, Base1 [22], Base2 [28], Base3 [4], and Base4 (FedAvg [17]) on LMR and SHL datasets.](image)

Figure 4 illustrates the impact of noise ratio on the global iterations (or communication rounds) using the LMR dataset. We fixed the value of desired precision $d_0 = 0.01$ and checked the efficiency of Fed-NL on different values of local epochs, i.e., $E = \{20, 40, 60\}$. The result shows that the increase in noise ratio increases the communication rounds, and a much higher increment is observed for the low value of local epochs. The result also demonstrates the communication round curve follows power-law till the noise ratio of 0.5 and shows a linear increment afterward. The model converges to a sub-optimal state when the noise ratio is more than 0.5. We further fixed the value of the local epoch to 20. We estimated the desired precision on a fixed number of communication rounds (100, 200, 300), as shown in Figure 5(b). We observe a similar pattern as Figure 5(a), where desired precision increases with noise ratio.

**Observation**: Noise ratio normalization and optimal weighted contribution are important factors to achieve high performance and minimize communication rounds. Using server dataset with noise-free labels mitigate the negative impact of the noisy labels in the dataset of the participants. Desired precision and local epochs play a decisive role while estimating the communication rounds for the convergence.

![Figure 5: Impact of noise ratio on global iterations (communication rounds) with fixed precision and local epochs on LMR dataset.](image)

Figure 5(a) illustrates the impact of noise ratio on the global iterations (or communication rounds) using the LMR dataset. We fixed the value of desired precision $d_0 = 0.01$ and checked the efficiency of Fed-NL on different values of local epochs, i.e., $E = \{20, 40, 60\}$. The result shows that the increase in noise ratio increases the communication rounds, and a much higher increment is observed for the low value of local epochs. The result also demonstrates the communication round curve follows power-law till the noise ratio of 0.5 and shows a linear increment afterward. The model converges to a sub-optimal state when the noise ratio is more than 0.5. We further fixed the value of the local epoch to 20. We estimated the desired precision on a fixed number of communication rounds (100, 200, 300), as shown in Figure 5(b). We observe a similar pattern as Figure 5(a), where desired precision increases with noise ratio.

**Observation**: Noise ratio normalization and optimal weighted contribution are important factors to achieve high performance and minimize communication rounds. Using server dataset with noise-free labels mitigate the negative impact of the noisy labels in the dataset of the participants. Desired precision and local epochs play a decisive role while estimating the communication rounds for the convergence.

7 CONCLUSION AND FUTURE WORK

In this paper, we proposed a federated learning approach in the presence of noisy labels in the participants’ datasets. Unlike the existing work, the Fed-NL trained the model on participants with noisy labels in their datasets. Fed-NL achieved significant performance by exploiting the server’s data instances with noise-free labels. We proposed an algorithm for the Fed-NL and performed its convergence analysis, followed by an estimation of the communication rounds for convergence. We also did the evaluations to verify the effectiveness of Fed-NL in the built environment.

We found the following conclusions from this work: federated learning work successfully only when the negative implications of noisy labels are effectively mitigated; correctly annotated server’s dataset can improve participants’ performance; the communication round depends upon the purity of the data annotation. Fed-NL technique considers the only noisy labels in the dataset of the participants. We will consider the noise in the input data instances along with labels. We also plan to consider different challenges, such as resource constraints and unbalanced datasets.

REFERENCES

[1] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra Perez, and Jorge Luis Reyes Ortiz. 2013. A public domain dataset for human activity. In Proc. ESANN. 437–442.

[2] Sebastian Caldas, Sai Meher Karthik Duddu, Peter Wu, Tian Li, Jakob Konečný, H Brendan McMahan, Virginia Smith, and Ameet Talwalkar. 2018. Leaf: A benchmark for federated settings. arXiv preprint arXiv:1812.01097 (2018).

[3] Pengfei Chen, Ben Ben Liao, Guangyong Chen, and Shengyu Zhang. 2019. Understanding and Utilizing Deep Neural Networks Trained with Noisy Labels. In Proc. ICLR. 1062–1070.

[4] Yiqiang Chen, Xiaodong Yang, Xin Qin, Han Yu, Biao Chen, and Zhiqui Shen. 2020. Focus: Dealing with label quality disparity in federated learning. arXiv preprint arXiv:2001.11359 (2020).

[5] Gregory Cohen, Sared Afsahr, Jonathan Tapon, and Andre Van Schai. 2017. EMNIST: Extending MNIST to handwritten letters. In Proc. IEEE IJCNN. 2921–2926.

[6] Mooming Duan, Duo Liu, Xianzhang Chen, Yujun Tan, Jingint Ren, Lei Qiao, and Liang Liang. 2019. Asteira: Self-Balancing Federated Learning for Improving Classification Accuracy of Mobile Deep Learning Applications. In Proc. IEEE ICCD. 246–254.

[7] Md Ouman Gani, Vaskar Raychoudhury, Janick Edinger, Valeria Mokrenko, Zheng Cao, and Ce Zhang. 2019. Smart Surface Classification for Accessible Routing through Built Environment: A Crowd-Sourced Approach. In Proc. ACM BuildSys. 11–20.

[8] Jiachao Gao, Wenzeng Peng, Zetian Liu, Md Farzay Rabbi Musam Billah, and Bradford Campbell. 2021. Decentralized federated learning framework for the neighborhood: a case study on residential building load forecasting. In Proc. Sensys. 453–459.

[9] Yufei Han and Xiangliang Zhang. 2020. Robust Federated Learning via Collaborative Machine Teaching. In Proc. AAAI. 4075–4082.

[10] Dan Hendrycks, Manta Mauricea, Duncan Wilson, and Kevin Gimpel. 2018. Using trusted data to train deep networks on labels corrupted by severe noise. In Proc. NIPS. 10456–10466.

[11] Alex Krizhevsky, Geoffrey Hinton, et al. 2009. Learning multiple layers of features from tiny images. (2009).

[12] Li Li, Huazhou Fu, Bo Han, Cheng-Zhong Xu, and Ling Shao. 2021. Federated Noisy Client Learning. arXiv preprint arXiv:2106.13239 (2021).

[13] Tian Li, Anit Kumar Sahai, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. 2020. Federated optimization in heterogeneous networks. Proc. NeurIPS 2 (2020), 429–450.

[14] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. 2019. On the Convergence of FedAvg on Non-IID Data. In Proc. ICLR. 1–26.

[15] Tongliang Liu and Dacheng Tao. 2016. Classification with Noisy Labels by Importance Reweighting. IEEE Transactions on Pattern Analysis and Machine Intelligence 38, 3 (2016), 447–461.

[16] Bing Luo, Xiang Li, Shiqiang Wang, Jianwei Huang, and Leandros Tassiulas. 2021. Cost-effective federated learning design. In Proc. IEEE INFOCOM. 1–10.

[17] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguerre y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In Proc. AISTATS. 1273–1282.

[18] R. Mishra, A. Gupta, H. P. Gupta, and T. Dutta. 2020. A Sensors based Deep Learning Model for Unseen Locomotion Mode Identification using Multiple Semantic Matrices. IEEE Transactions on Mobile Computing 20(2020), 1–10. doi: 10.1109/TMC.2020.3015546.

[19] Raed Abdel Sater and A Ben Hamza. 2016. A federated learning approach to anomaly detection in smart buildings. ACM Transactions on Internet of Things 2, 4 (2021), 1–23.

[20] Felix Sattler, Tim Korjakow, Roman Rischke, and Wojciech Samek. 2021. FEDAUX: Leveraging Unlabeled Auxiliary Data in Federated Learning. IEEE Transactions on Neural Networks and Learning Systems (2021), 1–13. doi: 10.1109/TNNLS.2021.3129371.

[21] SFL Challenge. 2022. http://www.sfl-dataset.org/activity-recognition-challenge/

[22] Tiffany Tuor, Shiqiang Wang, Bong Jun Ko, Changchih Liu, and Kin K. Leung. 2021. Overcoming Noisy and Irrelevant Data in Federated Learning. In Proc.
[23] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H. Vincent Poor. 2020. Tackling the Objective Inconsistency Problem in Heterogeneous Federated Optimization. In Proc. NIPS. 7611–7623.

[24] Lixu Wang, Shichao Xu, Xiao Wang, and Qi Zhu. 2021. Addressing class imbalance in federated learning. In Proc. AAAI. Vol. 35. 10165–10173.

[25] Shiqiang Wang, Tiffany Tuor, Theodoros Salonidis, Kin K Leung, Christian Makaya, Ting He, and Kevin Chan. 2019. Adaptive federated learning in resource constrained edge computing systems. IEEE Journal on Selected Areas in Communications 37, 6 (2019), 1205–1221.

[26] Yihao Xue, Chaoyue Niu, Zhenzhe Zheng, Shaojie Tang, Chengfei Lyu, Fan Wu, and Guihai Chen. 2021. Toward Understanding the Influence of Individual Clients in Federated Learning. In Proc. AAAI. 10560–10567.

[27] Haibo Yang, Minghong Fang, and Jia Liu. 2021. Achieving Linear Speedup with Partial Worker Participation in Non-IID Federated Learning. Proc. ICLR (2021), 1–23.

[28] Seunghan Yang, Hyoungseob Park, Junyoung Byun, and Changick Kim. 2022. Robust Federated Learning with Noisy Labels. IEEE Intelligent Systems (2022), 1–10. doi: 10.1109/MIS.2022.3151466.

[29] Xiyu Yu, Tongliang Liu, Mingming Gong, Kayhan Batmanghelich, and Dacheng Tao. 2018. An efficient and provable approach for mixture proportion estimation using linear independence assumption. In Proc. CVPR. 4480–4489.