Vol. 3 No. 2 (2025)

Spectrum of Engineering Sciences

SPECTRUM OF ENGINEERING SCIENCES

Online ISSN

3007-3138

Print ISSN

3007-312X

A Reliable Federated Learning Approach in Edge Computing

Amal Abdullah Mohammed Yayah[1*]

Computer Science and Technology Department, College of Computer Science and Electronic Engineering, Hunan University, Changsha, China. E-mail: yayahamal@hnu.edu.cn

Anwr Hasan Yahya Abohadi[2]

Computer Science and Technology Department, College of Computer Science and Electronic Engineering, Hunan University, Changsha, China.

Anwar Hasan Abo-hadi[3]

Faculty of Computer and Information Technology, Sana'a university ,Sana'a, Yemen

Abstract

The federated learning scheme proposes keeping data at the edge nodes, and bringing a key part of model training to the edge. So, in federated learning, edge parties maintain their own data and train the model in a distributed manner. Gradients or model updates are taking place between the edge participants and the centric-aggregator. Therefore, there is need to improve the processes that compose the federated learning training steps so that the system will gain better selection for participants in the process and apply an appropriate aggregation strategy that lets the global trained model's accuracy improve and takes advantage of the data diversity, especially in heterogeneous environments that are more common in reality. Adopting a federated learning system requires consideration of potential negative aspects. As long as a federated learning system more probably operates in an open environment that belongs to different organizations or individuals, that means the system is subjected to both honest participants that work properly and do the right duties in the process well, and dishonest participants that intentionally desire to affect the system and degrade the FL performance or get advantage from the system information. A reputation-based scheme for selecting trustworthy end nodes to participate in the federated learning process to be selected mainly in terms of measuring dataset overlaps, beside the embedded reputation that resulted from the contributions and integrity of participants in previous tasks with the owner, can be combined as an additional reputation term in case other cooperation between task owner and participant has taken place before

Keywords: Machine learning, Edge computing, Global model, Horizontal Federated learning

Introduction

In the current day, we reside in a time renowned for its plethora of data. This means that each aspect of our environment is interconnected with a source of data, and every facet of our lives is documented in a digital format. The present digital landscape encompasses a wide array of diverse types of data, including data from the Internet of Things (IoT), smart city data, cybersecurity data, smartphone data, health data, social media data, and numerous more[1].

Computers have the capability to execute a vast array of intricate and beneficial operations. Software applications can be developed to facilitate online communication, handle large amounts of data in databases, simulate and generate innovative discoveries in fields like science, engineering, and medicine, regulate and document industrial machinery, and even outperform skilled human players in games like chess. The functionality of these programs depends entirely on the creativity of their creators.

Machine learning is currently transforming various domains of science, engineering, business, and society. Machine learning focuses on constructing models that acquire knowledge from data in order to achieve high performance on a certain objective, rather than relying on predetermined logic. Machine learning systems necessitate increasing access to larger volumes of data to enhance their capacity to generalize to novel inputs. In today's world, there exists a plethora of data sources that could be employed to train models of machine learning. However, growing apprehension over data ownership and privacy has spurred the development of methods to preserve privacy in machine learning. In addition, the power, connectivity, and overall quantity of devices on the internet are growing rapidly, creating possibilities to transfer costly processing from the central to the periphery. This work focuses on federated learning, a revolutionary and modern approach to collaborative, distributed machine learning, while also safeguarding the data privacy of users.

Federated Learning

Federated machine learning involves multiple parties or participants working together to collectively train a single model while guaranteeing that the data of each client is kept private and is not

Spectrum of Engineering Sciences
Online ISSN
3007-3138
Print ISSN
3007-312X
SPECTRUM OF ENGINEERING SCIENCES

shared with other parties or a centralized server. The concept of federated machine learning could denote the practice of distributed learning involving Several participants too.

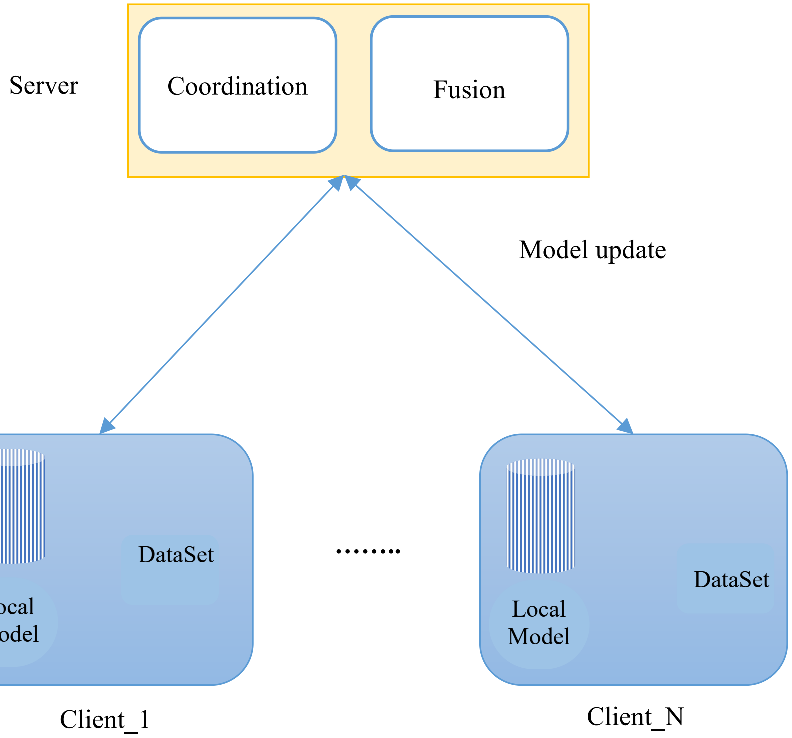


Fig. 1. General Architecture of Federated Learning

In this methodology, every participant undergoes individual training of a model using exclusively their own data, and thereafter sends their models. Once the model has been updated by all participating clients, the central aggregator combines those updated models sent by participants into a unified and updated global model version. This procedure is performed iteratively until the merged model reaches a certain threshold. Federated machine learning enables the preservation of privacy in machine learning by ensuring that sensitive and secret data remains under the authority of its initial proprietors.

This is achieved with local storage of data, which minimizes data movement among the parties involved in this process.

Technical Perspectives on Federated Learning

Federated learning scheme is an up-and-coming machine learning method that emphasizes privacy by safeguarding sensitive data. Additionally, it improves model performance by providing access to a larger or diverse data pool, thereby increasing the potential diversity of training data. A federated machine learning environment typically has four key sets of entities, as illustrated in Fig. 1, the primary server and so-called aggregator, the parties, which can be called clients or participants, the communication framework, and the aggregation method. Each of these components fulfills a distinct function in the process of federated learning. Those components can be described as the following:

Server: is the entity that oversees the connections between the components in the environment of federated learning (FL) and consolidates the knowledge generated by the FL participants.

Clients: That could include all computing devices that have sufficient computing resources and possess suitable data for training the model, including but not limited to servers, personal computers, smartwatches, cellphones, computerized sensor devices, and a variety of others.

The Communication Framework: It refers to the collection of devices and tools utilized to establish connections between servers, aggregators, and participants. This framework could take the form of the internet, an intranet, or an internal network.

Aggregation Method: the component accountable for combining the knowledge acquired by the participants through training the model with their own individual data, then updating the global model utilizing those obtained knowledges.

Federated Learning Types

Based on the data feeding the trained model and the participants involved in the FL system process, systems can be classified into several categories, primarily into the following two terms:

- **Cross-silo vs. cross-device system**
- **A horizontal vs. vertical system**

The cross-silo type of federated learning refers to the model training using distributed data that is located across various regulatory, organizational, or functional barriers. Typically, the data of those type of entities is stored in larger scala computing devices like bare metal servers or cloud instances, resulting in a relatively small number of training sets or silos. On the other hand, cross-device federated learning involves training models at the edge, directly on IoT devices such as drones, cell phones, or other similar systems. This scheme requires a large number of devices for the federation to be effective. However, it is limited by the low computational capabilities of individual devices and the increased likelihood of devices being offline and subsequently unable to be involved in the training process.

Related Work

The first federated learning algorithm is presented by google, based on a random participant selection model, known as Federated averaging (FedAVG). The primary objective of Federated averaging approach was cost achieving the desired accuracy while reducing the communication cost. Several experiments were conducted by the authors with different datasets such as CIFAR10, MNIST, and so forth in both independent- and -identically distributed (IID) and independent and identically distributed (non-IID) settings. The authors concluded that FedAVG can train models with good accuracy in a relatively a smaller number of communication rounds.

A significant issue with the aforementioned random client selection model and similar approaches that client dropout during FL training. It occurs when clients with limited computational resources are picked for training, leading to an ineffective training process. The longer update and upload time required for these clients can lead to delays in the overall training process. As a result, there is a need for more efficient client selection models that take into account, to handle the client heterogeneity and client dropout challenges of random selection, the global server takes into account the various client resources when choosing clients for FL training in resource-based selection. There are several resource-based client selection models introduced, that entails selecting clients based on various factors such as computing power, channel status, data size, CPU, memory, and energy consumption.

For instance, Abdulrahman et al. proposed Federated learning with Multicriteria Client Selection Model (FedMCCS) protocol, which selects clients based on CPU, memory, energy usage, and predicted download, update, and upload times[2]. In their work Nishio and Yonetani , Authors introduce Federated client selection protocol, which it's extension for federated averaging protocol for heterogenous environment [3]. This Federated client selection protocol (FedCS) tackles the issue of random client selection in federated learning process for cellular networks. Recognizing that clients possess varying computing resources, FedCS selects participants based on their ability to complete training within a defined timeframe. This resource-aware approach aims to accelerate model training step while achieving high-quality model which resulted in faster convergence model. other consideration is taking in the account the reputation of participators clients in the federated learning process, Song et al. introduced a reputation model the authors in their contribution, introduced model based on the beta distribution function, distinguishing between positive and negative behaviors[4] . Wang et al . in their work tested the participants before the model was trained and developed the participants' reputation directly from the test performance [5] . So, in this work, A reputation-based approach for selecting trustworthy clients to involved in the federated learning phases to be elected in terms of measuring dataset overlaps, beside the embedded reputation that concluded from the contributions and integrity of clients in previous tasks with the same or other task's owner, can be combined as an additional term in case another task between the publisher and client has taken place.

Model

The main workflow steps of federated learning mechanism are depicted in Fig. 2, the description of major steps will be described in the following sections.

Initialization

Participants and task owner engage in the process by registering, in such mechanism can be make the process transparent and confidential, such as blockchain accounts, ensuring the validity of their certificates of identity, and subsequently generating unique wallet addresses. These actions lay the foundation for their participation in blockchain- based transactions, facilitating a seamless interaction within the decentralized network.

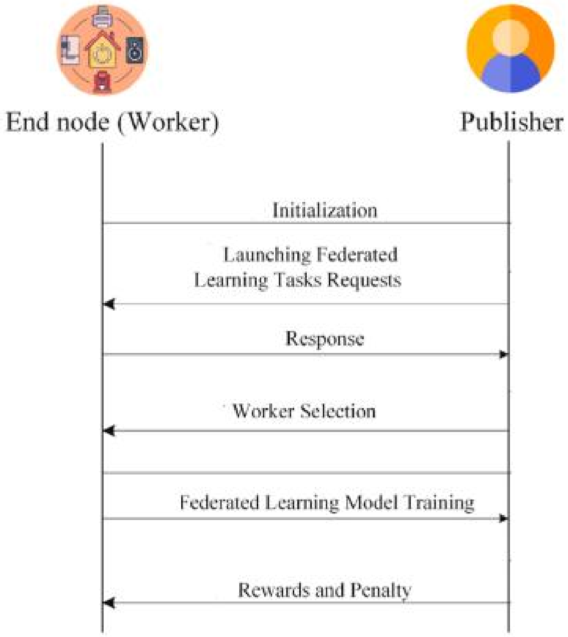


Fig. 2. The Fl Mechanism Workflow

By establishing these accounts and addresses, individuals can securely and transparently conduct various tasks and transactions, leveraging the inherent benefits of blockchain technology such as immutability, security, and decentralization. In this work experiments we will focus in the federated learning itself assuming that other arrangements such, management arrangements and privacy issues of the operation are fulfilled.

Retrieval of the Model

When a publisher which is the task owner s has a need for the model, $\boldsymbol{s}$ initially query the blockchain verifying if the model has been either previously trained or still. If there is an exist record for that model, s will directly conduct a model transaction with the original model's owner, Alternatively, $\boldsymbol{s}$ will initiate a task for FL training in smart contract form.

Launching FL Tasks Request

Task owner s publish the task broadcasting a detail of smart contract including the FL task requirements. That requirements could include identifier of task ID_S, types of data D_s, attributes of data DA_s , participant selection time T_s, data providers' total points τ_s that expected to involve participant in the training and rewards R_s , ….. etc. After task owner issue the task contract, participants who willing to involved in the training process will submit a request to s, which will include the type and attribute of their data as well as any necessary information for the task owner. Following collection period, s obtains a total request of involving from |P | participants, in that group of participants is indicated as P, $p_i \in$ P. D_i and DA_i are the data type and attribute of p_i respectively

Participant Selection

Task owner s will carefully identify appropriate candidates among the pool of available participants for training, concurrently, as a part of managements operation; calculations could be conducted to determine the required upfront deposit of points from each selected participant to involve in the process of FL training. This process is one step that help to ensure that more qualified individuals are chosen and adequately prepared for the training ahead. Additionally, meticulous consideration will be given to factors such as skill level, experience, and availability, to optimize the effectiveness of the training initiative while fostering a conducive learning environment for all participants.

The selection process of participants that will be involved in FL training process is primarily divided into two stages. The first stage is to involve build a pre-selected nominee set of participants according to kind of reputation based on chosen criteria that mainly purpose to get more reliable data as possible seeking gain higher accuracy, that comes mainly with find good matching with the requirements of the task. The second step as more for managements, transparent and privacy issues is to finalize involving the participants in the task of FL and to ensure lock the deposit points of the participants that's required for this the task.

FL Model Training

In the realm of federated learning (FL), the collaborative effort of selected participants is paramount in the iterative refinement of the model, a process that is central to the success of decentralized model training paradigms. Throughout multiple rounds of training, each participant contributes to the collective advancement of the FL model through a series of meticulously orchestrated steps.

Firstly, in the stage of local model training, individual clients independently undertake the responsibility of training the FL model using their respective allocated data shards. This crucial phase involves a sophisticated interplay of forward and backward propagation iterations, wherein model parameters are continuously adjusted to minimize the loss function. Through this localized training process, each client leverages its unique dataset to contribute valuable insights and updates to the global model and this strategy modify the local model training by adding penalize value to loss function as mentioned in a later section.

Subsequently, in the stage of model aggregation, the updated model parameters from each participant are harmoniously aggregated to generate a unified global model update. The disparate local updates are seamlessly merged while meticulously safeguarding the privacy and confidentiality of individual client data. This collaborative aggregation process ensures that the collective knowledge and expertise of the involved participants are effectively harnessed to enrich the global model, and we make it far away better by using the proposed aggregation technique as averaging for max accuracy in case of classification problem of local model as mentioned in later section.

Reliable Mechanism

To accomplish robust and reliable Federated learning process in such environment like mobile edge computing, some improvement steps have proposed. To boost reliability, a reputation and data-based participant’s selection scheme is presented. Defense mechanism is integrated to mitigate the effect of poisoning attacks on the performance of model by detecting poisoning attacks then a defense against these attacks. To increase model convergence then accuracy in more realistic scenario with non-iid a strategy of aggregation is implemented.

Data-Based And Reputation Client Selection

Select a suitable participant in Federated learning process is one of key step for reliable FL process as the following:

Initial Selection Of Participants Based On Initial Data

After getting responses from P, participants, s will perform a pre-liminary process for selection clients based on reputation. To ensure that we need that participants be go in multistep process to detects specific things to achieve the goal to find combined for better reputations and data-based client selection. Firstly, the data type of the participant is required to meet the requirements of task owner data type, that means, $D_s = D_i$, here we ensure that FL client have the same kind of data that we need to train our model.

In second step for data attributes , if the candidate participants fulfilled all attributes asked by task owner , in case that there are no identical similarities in the attributes of data , actually this cases is the more reality scenario , there is specific threshold for the disparity between the provided attributes of the participants and the attributes that is needed by s, could be used, to achieve that here, the Jaccard distance principle is utilized to quantify the distance between two sets of attributes [6] .

$$\frac{DA_i \cap DA_s}{DA_i \cup DA_s} \leq \theta s \qquad .. 2$$

Where DA_i , DA_s represent data attributes in participant and task owner respectively, and θ_s is a threshold, which is typically established by owner s depending on experience or other minimum certain criteria for the task owner, it's meaning the minimum requirement or availability of data within P, the set of participants that are indicated as having passed the initial requirement is denoted as P^{pre}

$$p_j^{pre} \in P^{pre}, 1 \leq j \leq |P^{pre}|. \qquad .. 3$$

After we make sure that users have the data, that the model need with attributes that at least get the threshold , It's need to investigate more the participants themselves for doing that , there are two additional steps could be done , first make sure from the reputations of participants because with participants that have higher reputation mostly have higher accuracy and more reliable data, these perform a vital role in the process of training the model [7]. Participant selection based on reputation value involved in first part with Direct- evaluation reputation. In direct- evaluation reputation of p_j^{pre} mainly take into account two parts, which are contribution of p_j^{pre} in the task that is in normal state is the dominant part and uncertainty of p_j^{pre} . With denote that $\alpha_j, \beta_j, \sigma_j$ represent times number that model of participant p_j^{pre} is chosen, unchosen during the FL training process in the historical task, and the communication quality factor respectively, so the contribution evaluation of this participant in history task with task owner s [8].

$$C_{s,j} = \sigma_j \frac{\alpha_j}{\beta_j + \alpha_j} \qquad .. 4$$

For simplify direct- evaluation reputation could be represented by the contribution of participant. Recommendation reputation evaluation that task owner s will also reflect the evaluation of p_j^{pre} that outcome from owners of other tasks over a period, which is could called-named as the reputation of recommendation evaluation. Assuming that all participants in this step have passed certain threshold for reputation to be ready for next step.

Participants Sortation Based On Data Availability (Jaccard Greedy Scheme)

The Jaccard Greedy Selection Strategy (JG)is used in this work approach to tackle the fundamental challenge of client selection within federated learning process, offering a good and comprehensive solution to a crucial aspect of distributed machine learning. As main part of this strategy lies on utilization of the Jaccard similarity metric, a powerful measure of overlap between datasets, which performs main role in the identification and prioritization of clients for participation in training during the FL process

$$J(Di, Ds) = \frac{Di \ \cap Ds}{Di \ \cup Ds} \qquad .. 5$$

Whereas Di, Ds represent numbers of data labels within i ,s respectively ,This metric functions as a quantitative gauge of the degree of similarity between a participant's dataset and the global dataset and in addition to that it sorts according to data labels that are available in client device, providing invaluable insights into the representativeness and relevance of potential participants. The algorithmic workflow of JG Strategy is characterized by its iterative and strategic approach to client selection. Beginning with the filtering and sorting of available participations based on their Jaccard similarity scores, the algorithm systematically identifies and prioritizes participations whose datasets exhibit the highest overlap with the global dataset. This procedure ensures participants inclusion of whose data distributions are more representative of the broader population, thereby improve the diversity and comprehensiveness of the federated learning process.

In addition to its foundational principles, the Jaccard Greedy Selection Strategy embodies a commitment to optimize and efficiency in participations selection. Through trying finding better clients that have best Jaccard similarity rating using a greedy algorithm and it used for making the locally optimal choice at each round communications, a greedy approach does not guarantee an optimal solution, but a greedy heuristic can get locally optimal solutions which is closely approximate a globally optimal

Algorithm 1. JG Selection Strategy

```
Input: $P^{can}, D_i^k, D_s^k, k = 1, 2, \ldots . N$ ,
$L \leftarrow \emptyset$
$P^{Selected} \leftarrow \emptyset$
$P^{Sorted} \leftarrow jaccardSimilaritySort(P^{can}, D_i^k, D_s^k)$
While L is not $= D_s^k$ do
  for: $i = 0, \ldots, k - 1$ do
      if $D_i^k \cap L \neq \emptyset$ do
       $L \leftarrow L \cup D_i^k$
       $P^{Selected} \leftarrow P^{Selected} \cup P_i^{Sorted}$
  end if
```

```
 end for
end while
function jaccardSimilaritySort($P^{can}$, $D_i^k$, $D_s^k$)
   for: $i = 0 , ..... , k - 1$ do:
   for: $j = 0 , ..... , k - 1$ do:
         if jaccardSimilarity ($D_{i,j}^k$ , $D_s^k$ ) $<$
 jaccardSimilarity ($D_{i,j-1}^k$ , $D_s^k$ ) do:
               Swap $P_{i,j}^{can}$ and $P_{i,j-1}^{can}$
         else
               break

         endif
      endfor
    endfor
    return $P^{can}$
function jaccardSimilaritySort($D_i^k$, $D_s^k$)
  intersection   ← $D_i^k$. intersection. ($D_s^k$)
  Union   ← $D_i^k$. intersection. ($D_s^k$)
  Similarity   ← |intersection| / |union|
  return  Similarity
```

solution within an a reasonable time amount [9] and in addition to don't dropped into local model we randomly shuffle the participants and apply Jaccard Greedy Selection strategy until it get good combination of clients that cover all data labels and in same time have more jaccard points

This strategy offering a sophisticated approach to client selection harnessing the power of jaccard similarity and integrating it into a systematic framework for client prioritization, this strategy not only enhances the diversity and representativeness of participants but also ensures the integrity and reliability of federated learning models across distributed datasets.

For the management part for getting more transparent and secure condition for the operation finalize the selection process take place, so after determining the range of selected participants and filtering them according to the previous steps, The clients highly susceptible to Sybil attacks from adversaries due to the public availability of participants reputation information on the blockchain. So, cryptographic sortition mechanism could be applied that, the system procedure operates by assigning the likelihood of a participant getting chosen proportional based on the points number they have in their wallet[10]. The participants themselves be able to determine whether they are chosen for the FL process through calculations. This process from task owner s side means finalize the involved participants, the same process from the participants perspective to confirm they are selected or not.

Local Training Strategy

The local training strategy functions a pivotal role in shaping the convergence, efficiency, and performance of decentralized model training across distributed nodes or devices. This section delves into the intricacies of local training methodologies, with a focus on its synergy with FedProx incorporating a penalizing term (mu). Local training involves the iterative optimization of model parameters on individual devices or nodes using local data samples. This decentralized approach is fundamental to federated learning, enabling privacy-preserving model updates without necessitating data centralization. However, the inherent challenges of non-IID data distribution and model heterogeneity underscore the need for innovative techniques to harmonize local updates and facilitate global model convergence.

As discussed previously, FedProx introduces a penalizing term (mu) to incentivize model coherence across local updates. This regularization mechanism encourages consensus among local models while allowing for adaptation to individual data characteristics. The incorporation of FedProx with a penalizing term within the context of local training yields several benefits, that could include enhanced model robustness, accelerated convergence, and improved resource utilization. The optimization objective in local training can be formulated as:

$$\left[minimize\ \frac{1}{n}\sum_{i=1}^{n} \mathrm{f}_i(\theta) + \frac{\mu}{2}\left|\theta_{i}^{rd} - \theta_{global}^{rd}\right|^{2} \right] \quad ..\ 6$$

Here, $\mathrm{f_i}(\theta)$ represents the local objective function of the i^{th} node, global model as θ_{global}^{rd} , for local model is θ_{i}^{rd}. The penalizing term enforces proximity between local and global model parameters $\left|\theta_{i}^{rd} - \theta_{global}^{rd}\right|^{2}$, thereby promoting model coherence. The local training strategy serves as the cornerstone of federated learning, dictating the efficacy and scalability of decentralized model training.

Through the integration of these useful techniques such as FedProx with a penalizing term, federated learning systems can achieve heightened convergence, efficiency, and privacy preservation, paving the way for advancements in various domains.

Strategy of aggregation

As part of training process of FL, one of primary steps for the process, straightforward and efficient aggregation mechanism is presented that is based on the basic approach of FedAvg but for preferable clients that we get from participants selection step. s will get the average parameters for those clients that have been selected before.

$$\theta^{rd} = \frac{1}{K}\sum_{\mathrm{k}\in\mathrm{St}} \left(\theta_k^{rd}\right) \qquad ..\ 10$$

Whereas K represent number of selected clients, θ_k^{rd} mode parameter for client k in specific communication round rd

Reliable FL Algorithm

Federated Learning is a method that involves training a model M in a collaborative manner to complete tasks assigned by a central entity, task owner referred to as s. various algorithm in machine learning could be used to train model M, such as Stochastic Gradient Descent (SGD), once the group of participants is established, these nodes, along with s, form a network shard. Each participant node p^{fin} initiates a transaction transfer corresponding points of deposit to the specified address of contract in accordance with the smart contract delineated by s. Subsequently, upon completion, these transactions. In the event a participant node fails to commence the transaction within the specified deadline, subsequent information dissemination to the non-compliant node will be halted. The details of this step and similar steps are not main focused in this work and treated as one of management section steps, following these preparatory steps, the process of FL starts.

Initiating the FL process, s dispatches the initial global model M^{0}_{global} to each participant node p_b^{fin} . Subsequently, each participant node selectively collects data pertinent to the task, to handle the training purposes of this task, Employing the acquired local data, each participant node p_b^{fin} proceeds to train a model M_b, transmitting the resultant model M_b back to s. Then s aggregates the updated model parameters from preselected participants by employing strategy in Eq (10), to serve as the global model for subsequent rounds, this equation signifies the selection process wherein the global model $\mathscr{M}$ is chosen based on its effectiveness. In summary, FL orchestrates a cooperative paradigm wherein a global model is iteratively refined through the collaborative efforts of distributed participant nodes. The utilization of machine learning algorithms, coupled with diligent privacy-preserving measures, ensures the integrity and efficacy of the collaborative training process. This scientific elucidation provides insights into the operational dynamics and algorithmic intricacies inherent in FL methodologies, that requires ongoing the continued work for further research and development in this important field of study

The procedure is iterated till the FL performance level reaches a specific threshold or until the time of training runs out. At the end, task owner gains the FL model. In the meantime, s would compute each participant contribution for those engaging in the training of this task according to the relevant contribution policy that will get conclude with reputation that it could use it in the future task in case another cooperation with those clients could be done for FL process

Results

The experiential study will answer about the how it this work algorithm accuracy in classification task, in different situations with independently and identically distributed setting, which is less common in the reality. And go farther to more reality scenarios with sitting non-iid, non-independently and identically distributed, even go farther with more complex classification with limited label number. To evaluate each one of the federated algorithms, we leveraged a dataset of Recognition of Human being Activity Using Smartphones (HAPT) and distributed this data through 50 clients, FL participants. From those clients, we selected 10 clients for training and evaluation of the global model. This section presents the results of an experimental study performed to evaluate the performance of some of the federated learning algorithms proposed in the literature, as well as this work approach. To demonstrate and comparing the effectiveness of the proposed mechanism in this work in term accuracy, HAPT classification datasets was built recording performing activities of daily living for 30 participants that include walking, lying, walking downstairs, standing, sitting, walking upstairs. while carrying embedded sensors within a waist-mounted smartphone that treated to contain the privacy information of the individual user [11] .

To demonstrate the versatility of our approach, we conducted separate experiments for both independent and identically distributed (IID) data and non-independent and identically distributed (non-IID) data scenarios. In practice, external factors like geographical locations result in clients only having access to local information, leading to significant label deviations among collected data, known as non-IID. When simulating the IID scenario, in term of the label numbers the training data is divided uniformly. Conversely, in simulating the non-IID scenario, the training set undergoes preprocessing, resulting in varying label distributions among different participants. The experiment setup includes various parameters tailored for a federated learning scenario. The experiment runs for 200 communication rounds with two epoch per round, indicating a relatively short training duration and making less communication overhead.

Basic Classification Task

Basic classification for dataset, independent and identically distributed, have done and we compare the results in this work with federated averaging, Proximal, and Max value algorithms with both datasets as the following:

Basic Classification Task IID Datasets on HAPT Dataset

Here we make the evaluation with HAPT dataset, plotting HAPT-IID with algorithms FedAvg, FedProx, FedMax, and this work as shown in Fig. 3. As the number of communication rounds rises, the level of accuracy gradually improves, noted that this work approach and is make convergence a little bit faster than other algorithms. In general,

Spectrum of Engineering Sciences
SPECTRUM OF ENGINEERING SCIENCES
Online ISSN
3007-3138
Print ISSN
3007-312X

with IID distribution the performance is highly similar. In this approach in IID setting dataset case, it has been seen that accuracy improves by around 0.4% compared to second algorithm behind FedAvg with accuracy around 94.3%.

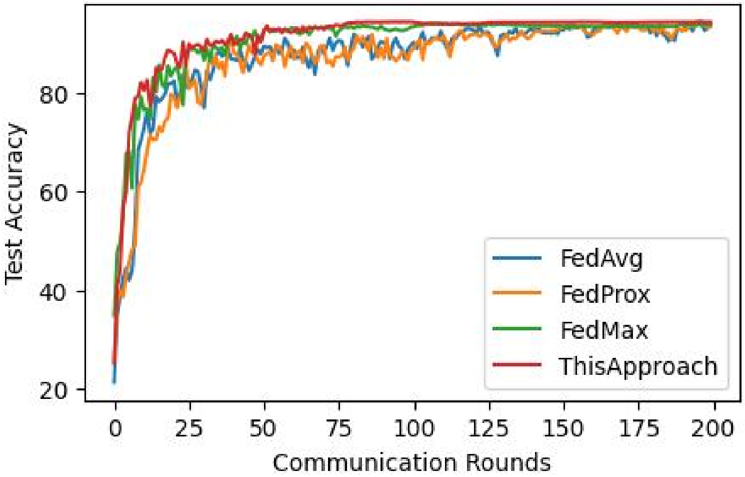


Fig. 3. Hapt Dataset – Experiment With Iid Setting, Fl Accuracy Performance Comparison

Complex Classification Task (Non-IID)

To manipulate complex classification tasks, we will use Non-Independent and Identically Distributed datasets for complex tasks is advantageous in such datasets offer a more realistic simulation of real-world scenarios where data is often distributed non-uniformly across different sources or locations. This realism is particularly relevant in applications like federated learning, where models are trained across distributed devices with varying data distributions based on user behavior, or device type.

From 1 to Half Label Distributions

We split dataset labels from 1 to 3 for HAPT distribution where randomly each class have different labels (1-half) with different distribution for this data as shown example of such distribution in Fig. 4, Starting to test this work algorithm on non-IID dataset with the other algorithms, FedAvg, FedProx, FedMax, with 200 round communications for HAPT and 10 participants .

Non-IID HAPT Dataset

Turning to the non-IID HAPT dataset, similar trends in accuracy are observed among the four algorithms, albeit with some variations. This work approach maintains its high performance with an accuracy around of 92.5%, surpassing FedAvg, FedProx, and FedMax as shown in Fig. 5. Again, the success of that can be attributed to its tailored client selection strategy, which ensures comprehensive coverage of all labels in the global model. This strategy proves effective in mitigating the difficulties posed by the HAPT dataset with non-IID distribution data setting. FedAvg and FedProx perform relatively well, with accuracies of 85.8% and 86.84%, respectively. These algorithms exhibit comparable performance but still fall a little bit short compared to achieved results. FedAvg relies on averaging, while

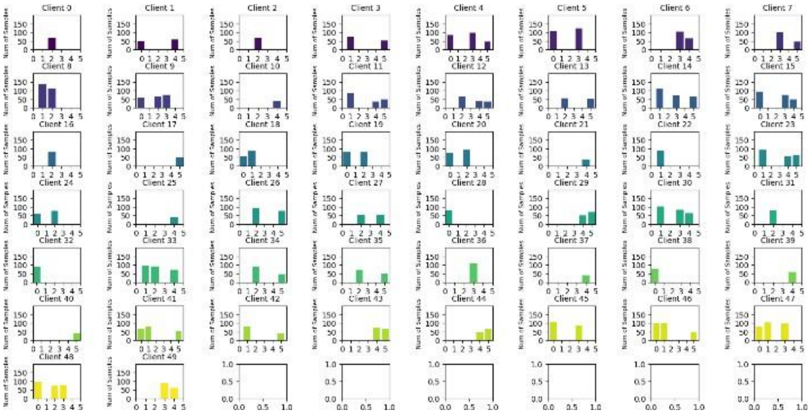


Fig. 4.non-IID 1 to 3 Label Distribution for HAPT dataset

Spectrum of Engineering Sciences
SPECTRUM OF ENGINEERING SCIENCES
Online ISSN
3007-3138
Print ISSN
3007-312X

FedProx incorporates a proximal term to enhance optimization. However, both algorithms little struggle to fully adapt to the diverse data distributions present in the HAPT dataset.

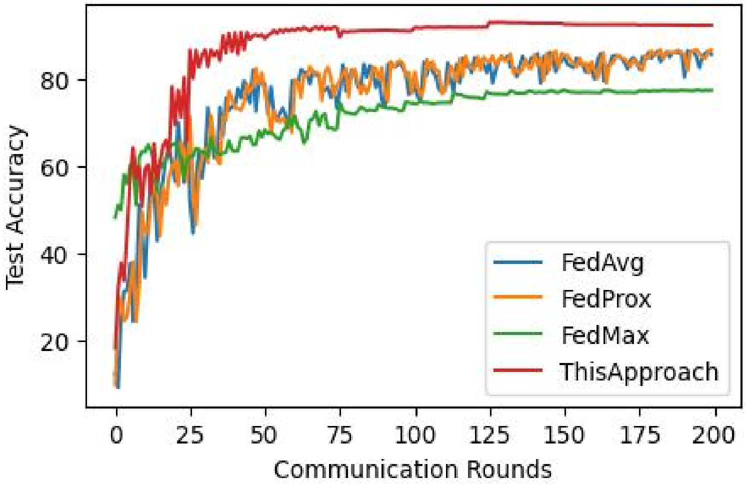
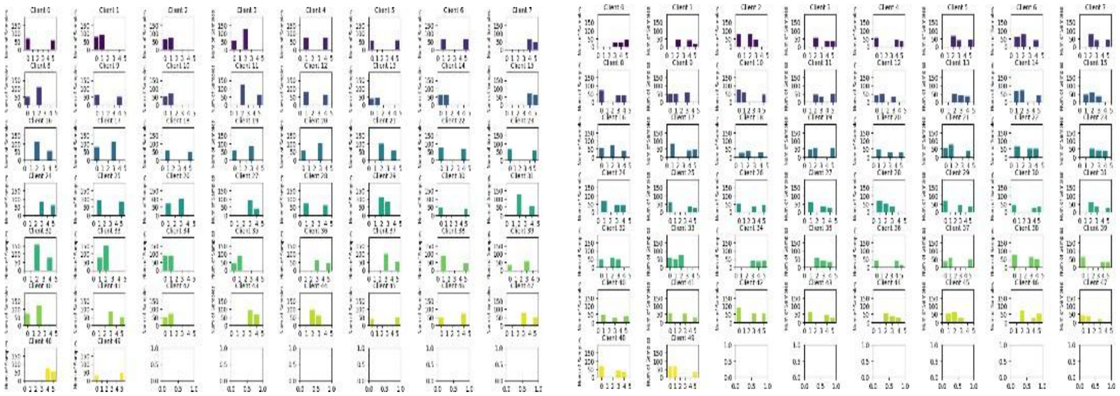


Fig.5. Non-iid data setting datasets- (1-Half) label distribution accuracy performance comparison

In contrast, FedMax shows a decrease in accuracy, achieving only around 77.5% , FedMax's reliance on the maximum client accuracy proves to be inadequate in handling the non-IID nature of the HAPT dataset. This algorithm fails to capture the variability and complexity of the data distribution across clients, leading to suboptimal updates of the global model

Different Levels Of Non-Iid Label Distributions

We extend the test to different levels of label distribution like 2 labels per client, 3 labels per client and 4 labels per client to test the ability working well at different of label distribution. The following Fig. 6, show example of the different 2 and 3 label distribution for datasets HAPT .



(a)

(b)

Fig. 6. Example of non-iid label distribution in HAPT : a) 2 labels ; b) 3 labels

When analyzing the results of such distribution for both HAPT dataset with non-IID data setting starting with a 2 labels distribution that show in Fig. 7., a noticeable drop in accuracy is observed across all algorithms. However, Our Approach demonstrates good resilience to this drop, maintaining relatively high accuracies compared to FedAvg, FedProx, and FedMax. For the non-iid HAPT dataset, this approach achieves an accuracy of 84.1%, outperforming the other algorithms significantly with this low number of labels.

This work through its sophisticated client selection strategy, which prioritizes clients based on the relevance and reliability of their data contributions make good tolerance to such hard distribution. By utilizing a greedy search approach on sorted clients, that relatively ensures that the right clients with the appropriate data and reputation are selected for model updates. This strategic selection process allows this approach to mitigate the negative impact of a reduced label distribution, as it can adapt dynamically to the available data sources and prioritize clients with diverse and representative data.

In contrast, FedAvg, FedProx, and FedMax exhibit significant drops in accuracy when the label distribution decreases to 2. These algorithms may struggle to effectively utilize the limited label information available from the clients, resulting in suboptimal model updates and decreased overall accuracy. FedAvg, for example, achieves an accuracy around of 75% in the non-IID setting HAPT

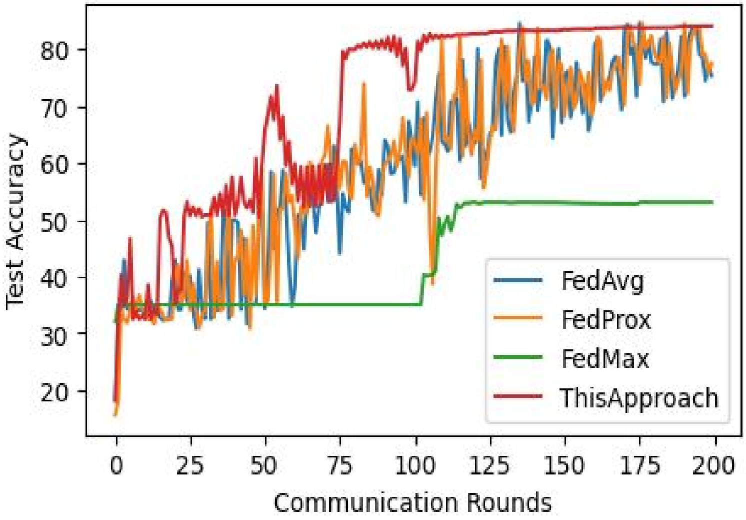


Fig. 7. non-iid data setting HAPT - (2) Label Distribution accuracy performance comparison

The drop in accuracy for these algorithms underscores the importance of adaptive and intelligent client selection strategies in federated learning settings, particularly when dealing with limited label distributions. That will keep relative ability to maintain high accuracies despite changes in label distribution highlights its effectiveness in selecting the right clients with the right data and reputation to contribute meaningfully to the global model update process.

When the label distribution increases to 3 as results shown in Fig. 8, This approach continues to demonstrate good performance, achieving accuracies of 93 % in the non-IID HAPT dataset. This further reinforces the efficacy of this approach in adapting to varying label distributions and leveraging diverse data sources to improve model performance. Comparing to FedAvg that archive around 88% and 81.4% in both datasets respectively . FedMax with such label number distribution archive 78.4 % and 56.2% that show its vulnerability to non-iid distribution especially with lower number of labels

In summary, this approach's strength lies in its adaptive client selection strategy, which allows it to maintain relatively high accuracies even in scenarios with reduced label distributions. This resilience underscores the importance of intelligent selection of participants involved in federated learning and highlights effectiveness in addressing the challenges posed by limited label information.

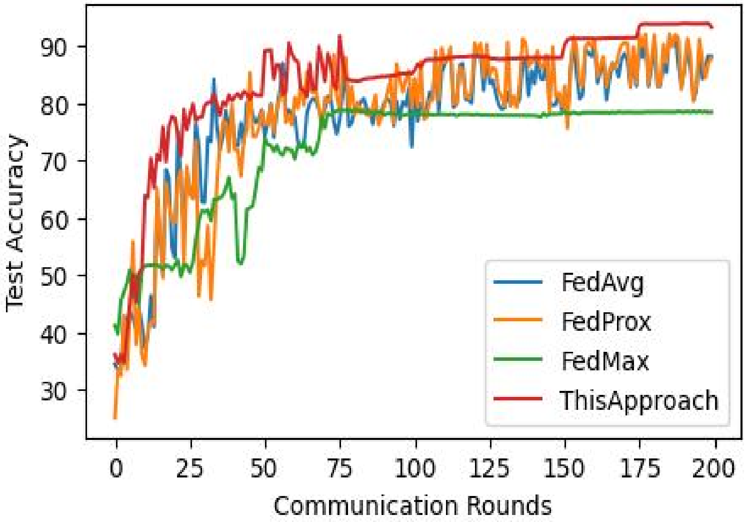


Fig. 8. Non-iid data setting HAPT - (3) Label- distribution accuracy performance comparison

With diversity of data pools among federated learning participants, especially with open and mixed environment such as edge computing that compose variety of IoT devices, getting more possible way for suitable selection that reflect the reliability is important. use mechanism that allow selection sub-part of clients that reflect their reputation in covering data labels that model will be trained on . This approach shows its effectiveness comparing to other shown approaches, FedAvg, FedProx, and Fedmax, especially with more realistic states in term of data distribution, non-iid setting and more with low number of labels. However, in dataset with iid-setting all mentioned approaches show nearly similar high performance.

Conclusion

With diversity of data pools among federated learning participants, especially with open and mixed environment such as edge computing that compose variety of IoT devices, getting more possible way for suitable selection that reflect the reliability is important. use mechanism that allow selection sub-part of clients that reflect their reputation in covering data labels that model will be trained on. This approach shows its effectiveness comparing to other shown approaches, FedAvg, FedProx, and Fedmax, especially with more realistic states in term of data distribution, non-iid setting and more with low number of labels. However, in dataset with iid-setting all mentioned approaches show nearly similar high performance.

References

1. Sarker, I.H., et al., Mobile data science and intelligent apps: concepts, AI-based modeling and research directions. Mobile Networks and Applications, 2021. 26(1): p. 285-303.
2. AbdulRahman, S., et al., FedMCCS: Multicriteria client selection model for optimal IoT federated learning. IEEE Internet of Things Journal, 2020. 8(6): p. 4723-4735.
3. Nishio, T. and R. Yonetani. Client selection for federated learning with heterogeneous resources in mobile edge. in ICC 2019-2019 IEEE international conference on communications (ICC). 2019. IEEE.
4. Song, Z., et al., Reputation-based federated learning for secure wireless networks. IEEE Internet of Things Journal, 2021. 9(2): p. 1212-1226.
5. Wang, Y. and B. Kantarci. A novel reputation-aware client selection scheme for federated learning within mobile environments. in 2020 IEEE 25th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD). 2020. IEEE.
6. Huang, X., et al., Distributed data trading algorithm based on multi-objective utility optimization. Journal on Communications, 2021. 42(2): p. 52-63.
7. Delgado-Segura, S., C. Tanas, and J. Herrera-Joancomartí, Reputation and reward: Two sides of the same bitcoin. Sensors, 2016. 16(6): p. 776.
8. Huang, X., et al., A reliable and fair federated learning mechanism for mobile edge computing. Computer Networks, 2023. 226: p. 109678.

9. Mestre, J. Greedy in approximation algorithms. in European Symposium on Algorithms. 2006. Springer.

10. Zhang, J., Y. Wu, and R. Pan. Incentive mechanism for horizontal federated learning based on reputation and reverse auction. in Proceedings of the Web Conference 2021. 2021.

11. https://www.kaggle.com/datasets/uciml/human-activity-recognition-with-smartphones, Human Activity Recognition with Smartphones.