FedLDCS: Adaptive Divergence-Based Client Selection for Federated Learning

Shiyue Hou
Northeastern University
Boston, USA
hou.shiy@northeastern.edu

Zhenglun Kong
Northeastern University
Boston, USA
kong.zhe@northeastern.edu

ABSTRACT

Federated learning (FL) revolutionizes machine learning by decentralizing data processing. It allows local devices to train models on their data and share updates with a central server, preserving privacy and optimizing bandwidth. Despite its potential, FL encounters challenges, especially in client selection, due to the non-independent and identically distributed (non-IID) nature of client data that can lead to performance deterioration, and the impracticality of engaging all clients simultaneously due to resource constraints and increased training expenses. To address these issues, we propose a novel *Largest Distance Client Selection* (LDCS) method that prioritizes clients based on the divergence of their local models from the global model, as quantified by the Frobenius norm. This strategy aims to optimize client participation by focusing on those with the most significant potential to enhance the global model, thereby improving training efficiency and model performance while overcoming the limitations of existing random or loss-based approaches. Experimental outcomes demonstrate that, in comparison with four existing client selection methods, our method achieves improvements of up to 5% and expedites the convergence process, with speed enhancements reaching as high as 8.5%.

KEYWORDS

Federated Learning, Resource Management, Client Selection

ACM Reference Format:
Shiyue Hou and Zhenglun Kong. 2024. FedLDCS: Adaptive Divergence-Based Client Selection for Federated Learning. In *FedKDD '24, August 26, 2024, Barcelona, Spain.*. ACM, New York, NY, USA, 6 pages.

1 INTRODUCTION

In the era of information, the rapid growth of data across various domains offers opportunities and challenges. A large volume of data is being generated by local devices, which can be utilized to obtain useful information for detecting, classifying, and predicting future events. Traditional centralized data processing model, which aggregates data from multiple sources on a single machine for analysis, is becoming less practical due to bandwidth constraints and rising concerns about data privacy and security. The growing awareness among individuals and organizations about the need to protect personal and sensitive information has underscored the importance of data security, making the acquisition of reliable and accurate data more challenging than ever. Federated learning (FL) emerges as a solution to these challenges, offering a decentralized approach to machine learning. One seminal work is FedAvg [6].

However, implementing the FL in real-life scenarios presents several challenges. While clients train models locally and share updates to improve the global model, not every client is (equally) beneficial for training the global model. One primary issue is the variability in data distribution across clients. Often, the training data on these clients are non-independent and identically distributed (non-IID), meaning that each device may have data that shifts in distribution, quantity, or type, rather than being uniformly distributed (e.g., the same number of data points for each class). This discrepancy can introduce challenges to model training, potentially reducing accuracy. Another issue is the computation cost. Due to expensive communication resources, engaging in the training process all at once is impractical. This limitation arises from several factors, such as limited bandwidth, processing capabilities, and energy costs. Addressing these challenges hinges on strategically selecting a sound subset of clients for participation, a decision crucial for maximizing performance while minimizing energy consumption.

A straightforward approach involves randomly selecting a specific number of clients k from the pool each training round (u.a.r.). However, this may not fully leverage unique contributions of local updates due to the diversity in client data distributions and hardware, potentially undermining the global model's performance. Recent work utilizing the losses computed from individual client models. In AFL [3], the server calculates these losses to generate probabilities that guide the selection of the next group of clients for training. POW [1] prioritizes clients that exhibit the highest local loss according to the current global model, while UCB-CS [2] employs a loss reduction index to identify the top clients for participation. However, loss-based methods can inadvertently introduce bias. By disproportionately favoring clients that significantly reduce global loss, these strategies risk overlooking the importance of data diversity, potentially leading to model overfitting. Moreover, focusing on high-loss clients can strain resources and limit scalability, as it fails to consider the full spectrum of clients and their varied data contributions.

In this paper, we propose a novel *Largest Distance Client Selection* (LDCS) method. We aim to optimize client participation in model training by prioritizing those with the most significant potential for improving the global model. Specifically, the server first requests the local model from each client. Then it employs the Frobenius norm to quantify the divergence between each local and global model. This divergence serves as a basis for ranking clients: those whose

FedKDD '24, August 26, 2024, Barcelona, Spain.

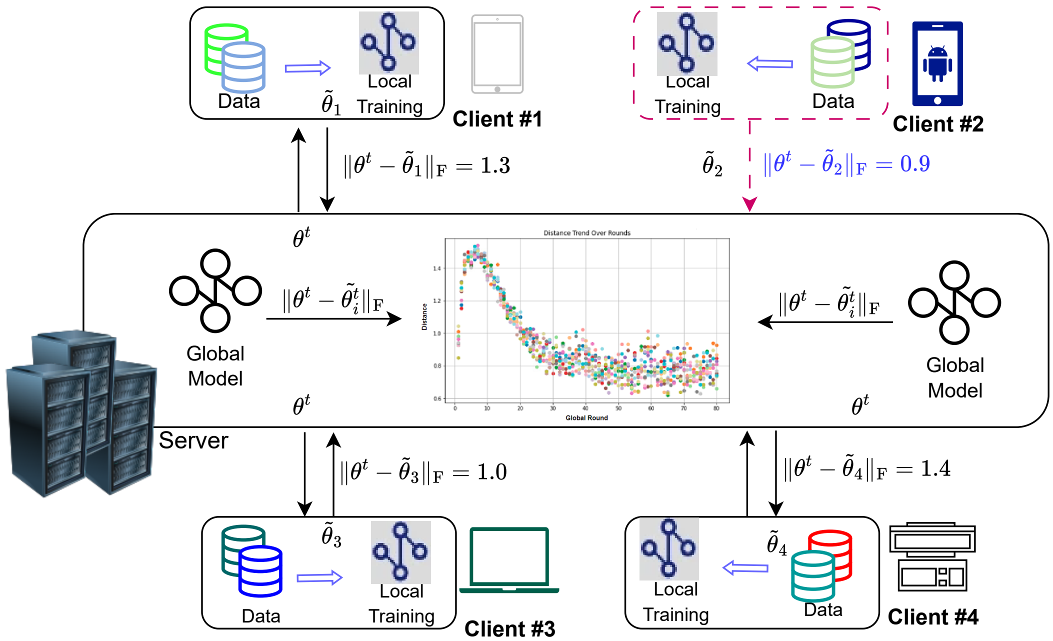


Figure 1: An illustrative example of FedLDCS, wherein 4 clients collaborate to train a model θ^t on non-IID datasets orchestrated by a server. An active client i optimizes the current global model θ^t on its local dataset and obtains an updated model $\widetilde{\theta}_i$. Client 2 is excluded due to its least divergence from θ^t. The solid lines between clients and the server denote bidirectional active communication links, whereas the dashed line refers to an idle link. $\|\cdot\|_F$ defines Frobenius norm.

local models exhibit large discrepancies from the global model are considered important. For subsequent training rounds, we select the top-k clients that demonstrate the greatest variance from the global model. This targeted selection criterion ensures that only clients with the most substantial local improvements are trained and updated. Consequently, this method accelerates training speed and sidesteps the inefficiencies associated with existing loss-based methods. This paper makes the following contributions:

- We propose a novel client selection methodology–Longest Distance Client Selection (LDCS) for non-IID federated learning.
- The novel approach enhances system efficiency by minimizing communication overhead through a strategic computation of the divergences between local and global models.
- In comparison to existing client selection methods, our proposed LDCS accelerates convergence speed by up to 8.5% and improves test accuracy by as much as 5%.

2 RELATED WORKS (OR BACKGROUND)

Resource optimization for federated learning is a common topic in recent research. Recent work has focused on the joint optimization of heterogeneous data, computation, and communication resources [7], [9], [11]. However, these approaches primarily aim to minimize computation times and/or energy consumption for general computational tasks, which differs significantly from our work. Our objective is to maximize the efficiency of training machine learning models. We assume a scenario where each client possesses data and computational resources, ensuring data privacy during ML tasks. These distinctions motivate our proposal of a new adaptive divergence-based client selection method for federated learning.

In [8], the authors introduced the Federated Learning Gaussian Process (FedGP) algorithm for client selection in heterogeneous scenarios. Their algorithm models the changes in client loss as a Gaussian Process (GP) to determine the selection strategy, resulting in fast convergence and reduced redundancy in client selection. However, their study only considered data distribution across the edge. In [10] and [12], client selection algorithms were introduced to account for heterogeneous data distribution across the edge, aiming to select edge devices that reduce model convergence time. In [4], the authors identified volatile clients—those unavailable for training during all rounds—and differentiated these clients in their selection criteria. They proposed a stochastic selection algorithm based on a fairness quota to balance convergence speed and model accuracy. However, their work did not consider energy consumption or its impact on client performance and the global model.

When client selection is based on features that do not accurately represent clients, some resources can be overused while others are underutilized. Utilizing a diverse set of features can lead to more efficient resource utilization, refining the selection of clients who significantly influence the model. In this paper, feature selection is based on metrics that more accurately represent edge devices, allowing us to explore the trade-off between resource usage and model performance.

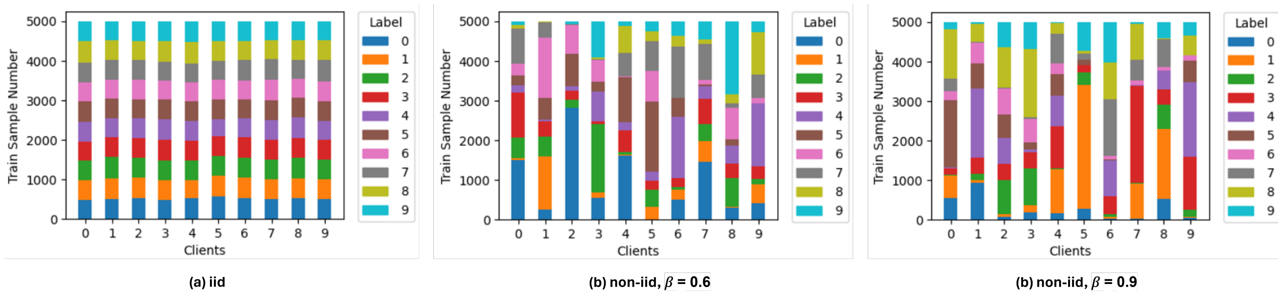


Figure 2: Label distributions under different Dirichlet distribution

3 MOTIVATIONS

Federated learning, by its very nature, introduces a distinct set of complexities when it comes to the selection of clients for model training. This is especially true in situations where data distributions are not independently and identically distributed (non-IID). Conventional approaches, such as random selection or sequential selection, frequently lead to inconsistent outcomes. Our research, carried out using a straightforward Convolutional Neural Network (CNN) model and the CIFAR-10 dataset across 10 clients under a variety of conditions, offers valuable insights into the selection patterns of these methods.

More specifically, we utilize the CIFAR-10 dataset, which is distributed under various Dirichlet distributions (i.e., non-IID datasets) and a uniform distribution (i.e., IID datasets), to assess different client selection strategies. For instance, Figure 2 illustrates the label distributions of the datasets with (a) a uniform distribution, (b) a Dirichlet distribution with $\beta = 0.6$ and 0.9. It is observable that different data distributions can impact the availability of labels per client, thereby influencing client selection strategies.

We then implement three conventional client selection strategies on datasets with varying data distributions: *Random*: Clients are selected randomly in each round of the learning process. *Random Round Robin*: The user selector selects users randomly in a round-robin manner. During each round, it uniformly selects users from those not yet chosen in the current epoch. *Importance Sampling*: User selector that performs Important Sampling. Each user is randomly selected with a probability proportional to (number of samples in user * clients per round) / total samples in dataset. These traditional strategies will also be used as the baselines for the comparisons in our evaluation.

From Figure **??**, it is evident that the performance of the importance sampling client selection algorithm surpasses that of random selection and outperforms random round-robin under IID datasets. The reason for this is that in an IID dataset, the equal distribution of labels allows importance sampling to prioritize clients with higher weights, thereby effectively enhancing learning in each training round. However, in a non-IID dataset, the skewed label distribution makes it challenging to identify high-weight clients, as the importance of each client's data is less clear.

Algorithm 1: FedLDCS

1 **Input:** Number of global rounds T, model initialization θ^0, number of clients N, number of local epochs E, number of selected clients K, local learning rates $\{\eta_t\}_{t=0,\cdots,T-1}$.
2 The server initializes parameter θ^0 and client i initializes $\widetilde{\theta}_i = \theta^0,\ \forall i \in [N]$. Define $\widetilde{\boldsymbol{\theta}} = \left\{\widetilde{\theta}_i \mid i \in [N]\right\}$;
3 **for** $t = 0, \cdots, T-1$ **do**
  /* On the server */
4   $\mathcal{S}_t \leftarrow$ LDCS$(\widetilde{\boldsymbol{\theta}}, \theta^t, K, t)$;
5   Broadcast θ^t to clients in $\mathcal{S}^t$;
  /* On the clients */
6   **for** $i \in \mathcal{S}^t$ **do**
7     $\widetilde{\theta}_i \leftarrow$ LocalOpt(θ^t, η_t, E);
8   Upload $\widetilde{\theta}_i$ to the server;
9   **end**
  /* On the server */
10   $\theta^{t+1} \leftarrow \frac{1}{|\mathcal{S}_t|} \sum_{i \in \mathcal{S}_t} \widetilde{\theta}_i$;
11 **end**
12 **Function** LDCS$(\widetilde{\boldsymbol{\theta}}^t, \theta^t, K, t)$:
13   $\mathcal{S}_t \leftarrow [N]$;
14   **if** $t = 0$ **then return** $\mathcal{S}_t$;
15   **else while** $|\mathcal{S}_t| > K$ **do**
16     $i \leftarrow \arg\min_{i \in \mathcal{S}_t} \|\theta^t - \widetilde{\theta}_i\|_F$;
17     $\mathcal{S}_t \leftarrow \mathcal{S}_t \setminus \{i\}$;
18   **end**
19   **return** $\mathcal{S}_t$;

4 THE PROPOSED METHODS

In this section, we introduce our proposed Federated Average with Largest Distance Client Selection (FedLDCS) in Algorithm 1. Specifically, we propose a novel client selection scheme termed as Largest Distance Client Selection in LDCS function.

Lines 5-10 illustrate a standard FL pipeline as in the FedAvg algorithm, where an active client i in the selected client set $\mathcal{S}_t$ downloads the latest global model θ^t from the server and optimizes on its local dataset to get an updated local model $\widetilde{\theta}_i$. Depending on the system configuration, local data distributions can either be IID or non-IID across clients. Departing from engaging clients uniformly at random, lines 11-18 describe the proposed LDCS module. In the first round ($t = 0$), all clients are required to participate in FL training to get the first update of $\widetilde{\boldsymbol{\theta}}$, a criteria to decide who to participate. When a client i is admitted to $\mathcal{S}_t$ and completes local training, the latest $\widetilde{\theta}_i$ replaces the staled one in $\widetilde{\boldsymbol{\theta}}$. Otherwise, $\widetilde{\theta}_i$ remains unchanged. Starting from the second round, an iterative procedure (lines 15-18) is implemented to pick the clients, whose models $\widetilde{\theta}_i$'s are the K farthest away from the current global model θ^t. The distance is measured by Frobenius norm. As shown, $\widetilde{\boldsymbol{\theta}}$ brings

Table 1: Comparisons of client selection schemes in terms of convergent round and test accuracy under different local data heterogeneity and different numbers of selected clients K out of 10 clients on CNN and CIFAR-10 [5]. The convergent round is defined as the round, where the fluctuations in test accuracy do not exceed 1% in a consecutive of the last 10 rounds.

Selection Scheme	Dirichlet Parameter β	Convergent Round			Test Accuracy		
		$K = 7$	$K = 8$	$K = 9$	$K = 7$	$K = 8$	$K = 9$
Random Selection	$\beta \to \infty$ (IID)	70	67	73	58.87%	60.07%	55.54%
	$\beta = 0.6$	72	69	77	56.87%	59.55%	62.46%
	$\beta = 0.9$	75	75	71	61.63%	57.69%	58.41%
Round Robin	$\beta \to \infty$ (IID)	86	89	87	58.55%	58.91%	58.07%
	$\beta = 0.6$	84	80	78	58.22%	58.08%	61.81%
	$\beta = 0.9$	86	76	89	61.94%	62.47%	60.46%
Importance Sampling	$\beta \to \infty$ (IID)	74	73	71	59.09%	59.10%	63.15%
	$\beta = 0.6$	70	73	79	62.66%	59.59%	64.13%
	$\beta = 0.9$	86	81	87	59.75%	62.40%	61.18%
Largest Distance (ours)	$\beta \to \infty$ (IID)	70	65	71	63.50%	60.08%	63.78%
	$\beta = 0.6$	74	68	70	62.08%	62.95%	65.95%
	$\beta = 0.9$	74	73	73	62.98%	62.89%	63.89%

in extra storage burden ($N \times d$ units of memory) to the server in particular due to the excluded clients in $[N] \setminus \mathcal{S}_t$, where N, d are the number of clients and the model dimension.

Let $\Delta_i \triangleq \theta^t - \widetilde{\theta_i}$ define the local improvement after client i's individual updates. At a high level, selecting clients that are with the most significant local improvement Δ_i's allows the server to emphasize the clients with the most potential while saving expensive communication resources. Intuitively, LDCS benefits a FL system with non-IID data more than one with IID data. When local data distributions are homogeneous, clients are likely to make alike improvements since their data samples are interchangeable. Therefore, the choice of participating clients might not be as important. In contrast, heterogeneous local data distributions entail dynamical local updates. Thus, it could be helpful to highlight only a subset of clients.

5 EXPERIMENTS

5.1 Implementation Details

In this section, we explore our experimental results. First, we overview the local data heterogeneity and experimental setups. Next, we discuss the results and implications for future directions.

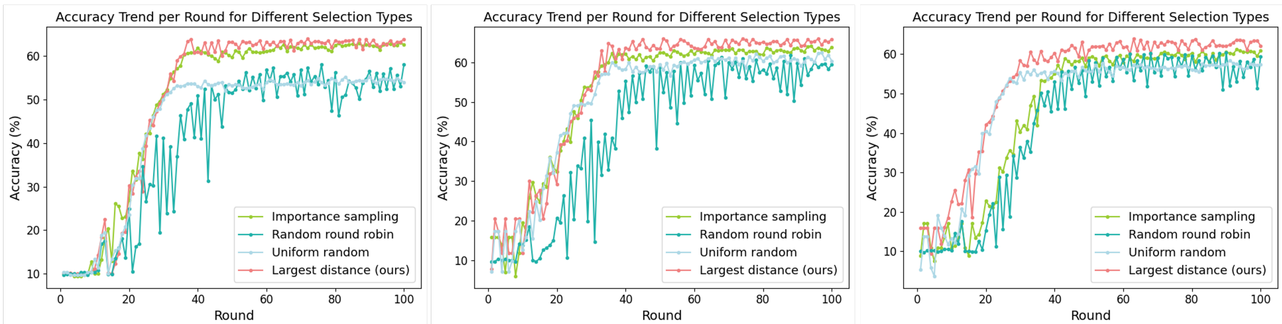
Local dataset and distributions. Our evaluation is an image classification task based on CIFAR-10 dataset [5]. We utilize `Dirichlet`(β) distribution to capture the fundamental non-IID distributions across clients. A smaller β implies a more heterogeneous local data distribution, and vice versa. Specifically, we consider $\beta \in \{0.6, 0.9\}$ in non-IID settings and $\beta \to \infty$ in IID setting, respectively.

Federated learning system. Our code is built on FLSim from Meta Research under Apache 2.0 License. All experiments are performed on a private computing cluster with 16 Intel Xeon W-2245 CPUs, 62GB of RAM, 1 NVIDIA Tesla P100 GPU on Ubuntu 20.04. In a total of 100 global rounds, one server and 10 clients collaborate to train a CNN model with only a subset of clients $\mathcal{S}_t$ admitted for aggregation in each round. Clients use mini-batch SGD with a batch size of 32 to perform local optimizations. The construction of $\mathcal{S}_t$ is at the server's discretion based on LDCS function in Algorithm 1.

5.2 Results and Discussions.

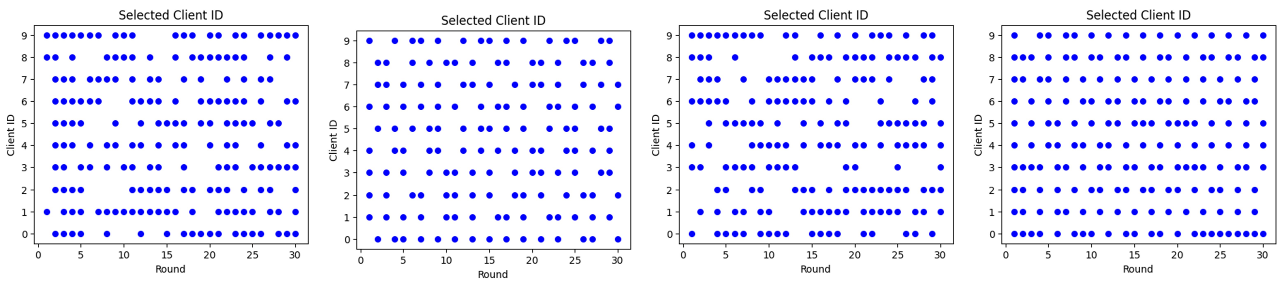
Table 1 provides a comparative analysis of four client selection schemes—Random Selection, Round Robin, Importance Sampling, and our proposed Largest Distance method—in terms of convergent round and test accuracy. The evaluation is performed under varying degrees of local data heterogeneity (represented by Dirichlet parameter β and different numbers of selected clients K from a total of 10 clients using CNN and CIFAR-10 datasets. We report both the test accuracy and convergent round.

The results demonstrate that under similar resource constraints, our proposed Largest Distance Selection method LDCS outperforms existing selection methods in terms of both convergent round and test accuracy in most tasks, particularly in non-IID data scenarios. Compared to Random Selection, our Largest Distance method consistently converges faster and achieves higher test accuracy across all values of β and K. For instance, with $\beta = 0.6$ and $K = 9$, our method converges in 70 rounds with an accuracy of 65.95%, whereas Random Selection requires 77 rounds and only achieves 62.46%. The Largest Distance method significantly outperforms Round Robin in terms of convergence speed and test accuracy. For $\beta \to \infty$ and $K = 8$, our method converges in 65 rounds compared to Round Robin's 89 rounds and achieves a higher accuracy of 62.08% versus 58.91%. When compared to Importance Sampling, our method shows superior performance in both convergence rounds and test accuracy. For $\beta = 0.9$ and $K = 9$, our method converges in 73 rounds



(a) iid (b) non-iid, $\beta = 0.6$ (c) non-iid, $\beta = 0.9$

Figure 3: Accuracy trend comparison of different selection types under elected clients K = 9 out of 10 clients.



(a) Important Sampling (b) Random Round Robin (c) Uniform Random (d) Largest Distance Selection

Figure 4: Patterns of different selection methods

with an accuracy of 63.89%, while Importance Sampling requires 87 rounds and achieves 61.18%.

In summary, the Largest Distance method demonstrates the best overall performance among the client selection schemes analyzed, offering both faster convergence and higher test accuracy, particularly under varying degrees of data heterogeneity.

Figure 3 illustrates the accuracy trends over 100 rounds for different client selection schemes under varying data heterogeneity conditions, with $K = 9$ out of 10 clients. The three subfigures correspond to different values of the Dirichlet parameter β, representing IID ($\beta = \infty$), non-IID ($\beta = 0.6$), and non-IID ($\beta = 0.9$) data distributions. The selection schemes compared are Importance Sampling, Random Round Robin, Uniform Random, and our proposed Largest Distance method. The accuracy trends depicted in indicate that our proposed Largest Distance method consistently outperforms the other client selection schemes across different data heterogeneity conditions. Its ability to achieve rapid and stable convergence to high accuracy levels, particularly in non-IID settings, underscores its effectiveness and robustness. The performance gap between the Largest Distance method and the other schemes is more pronounced in non-IID scenarios, highlighting its potential for federated learning environments where data heterogeneity is a significant challenge.

As shown in Figure 4 , the pattern of client selections varies significantly among the different algorithms. The largest distance selection method results in a much more diverse set of selected clients. In contrast, both random round-robin and random selection methods tend to concentrate on a limited subset of clients. This lack of diversity in client selection greatly affects the performance of the training process and explains why the largest distance selection method outperforms the other three algorithms.

6 CONCLUSIONS

In this paper, we propose a novel Largest Distance Client Selection (LDCS) scheme for federated learning aimed at boosting training performance caused by the non-IID nature of client data. Our method prioritizes clients based on the divergence of their local models from the global model. It seeks to optimize client participation by focusing on those with the most potential to enhance the global model. Experimental results show that our method improves training efficiency and model performance compared to existing methods.

REFERENCES

[1] Yae Jee Cho, Samarth Gupta, Gauri Joshi, and Osman Yağan. 2020. Bandit-based communication-efficient client selection strategies for federated learning. In *2020 54th Asilomar Conference on Signals, Systems, and Computers*. IEEE, 1066–1069.
[2] Yae Jee Cho, Jianyu Wang, and Gauri Joshi. 2020. Client selection in federated learning: Convergence analysis and power-of-choice selection strategies. *arXiv preprint arXiv:2010.01243* (2020).
[3] Jack Goetz, Kshitiz Malik, Duc Bui, Seungwhan Moon, Honglei Liu, and Anuj Kumar. 2019. Active federated learning. *arXiv preprint arXiv:1909.12641* (2019).
[4] Tiansheng Huang, Weiwei Lin, Li Shen, Keqin Li, and Albert Y Zomaya. 2022. Stochastic client selection for federated learning with volatile clients. *IEEE Internet of Things Journal* 9, 20 (2022), 20055–20070.
[5] Alex Krizhevsky, Geoffrey Hinton, et al. 2009. Learning multiple layers of features from tiny images. (2009).
[6] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*. PMLR,

1273–1282.
[7] Stefania Sardellitti, Gesualdo Scutari, and Sergio Barbarossa. 2015. Joint optimization of radio and computational resources for multicell mobile-edge computing. *IEEE Transactions on Signal and Information Processing over Networks* 1, 2 (2015), 89–103.
[8] Minxue Tang, Xuefei Ning, Yitu Wang, Yu Wang, and Yiran Chen. 2021. Fedgp: Correlation-based active client selection for heterogeneous federated learning. *arXiv preprint arXiv:2103.13822* (2021).
[9] Yongfeng Wei, Shanguo Huang, Shiyue Hou, Xinlu Gao, and Wanyi Gu. 2015. Tunable and integrated RF photonic phase shifter based on phase modulation and FBG filtering. In *AOPC 2015: Optical Fiber Sensors and Applications*, Vol. 9679. SPIE, 274–277.
[10] Hongda Wu and Ping Wang. 2022. Node selection toward faster convergence for federated learning on non-iid data. *IEEE Transactions on Network Science and Engineering* 9, 5 (2022), 3099–3111.
[11] Yinghao Yu, Jun Zhang, and Khaled B Letaief. 2016. Joint subcarrier and CPU time allocation for mobile edge computing. In *2016 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 1–6.
[12] Xukai Zhang, Junqi Wang, Shiyue Hou, Shanguo Huang, Wanyi Gu, et al. 2013. Four-wave mixing based phase-conjugation technique for retro-directive array. (2013).