CriticalFL: A Critical Learning Periods Augmented Client Selection Framework for Efficient Federated Learning

Gang Yan
gyan2@binghamton.edu
SUNY-Binghamton University
Binghamton, NY, USA

Hao Wang
haowang@lsu.edu
Louisiana State University
Baton Rouge, LA, USA

Xu Yuan
xu.yuan@louisiana.edu
University of Louisiana at Lafayette
Lafayette, LA, USA

Jian Li
lij@binghamton.edu
SUNY-Binghamton University
Binghamton, NY, USA

ABSTRACT

Federated learning (FL) is a distributed optimization paradigm that learns from data samples distributed across a number of clients. Adaptive client selection that is cognizant of the training progress of clients has become a major trend to improve FL efficiency but not yet well-understood. Most existing FL methods such as `FedAvg` and its state-of-the-art variants implicitly assume that all learning phases during the FL training process are equally important. Unfortunately, this assumption has been revealed to be invalid due to recent findings on critical learning periods (`CLP`), in which small gradient errors may lead to an irrecoverable deficiency on final test accuracy. In this paper, we develop `CriticalFL`, a `CLP` augmented FL framework to reveal that adaptively augmenting exiting FL methods with `CLP`, the resultant performance is significantly improved when the client selection is guided by the discovered `CLP`. Experiments based on various machine learning models and datasets validate that the proposed `CriticalFL` framework consistently achieves an improved model accuracy while maintains better communication efficiency as compared to state-of-the-art methods, demonstrating a promising and easily adopted method for tackling the heterogeneity of FL training.

CCS CONCEPTS

- **Computing methodologies → Distributed algorithms**.

KEYWORDS

Federated Learning, Critical Learning Periods, Client Selection

ACM Reference Format:
Gang Yan, Hao Wang, Xu Yuan, and Jian Li. 2023. CriticalFL: A Critical Learning Periods Augmented Client Selection Framework for Efficient Federated Learning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '23), August 6–10, 2023, Long Beach, CA, USA.* ACM, Long Beach, CA, USA, 10 pages. https://doi.org/10.1145/3580305.3599293

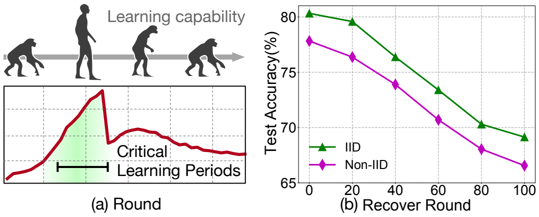


Figure 1: (a) An intuitive example of `CLP`. (b) The final accuracy achieved by ResNet-18 on both IID and Non-IID CIFAR-10 with `FedAvg` for training as a function of recover round, before which only partial datasets are used [52].

1 INTRODUCTION

Federated learning (FL) [32] has emerged as an attractive distributed learning paradigm that leverages a large number of clients to collaboratively learn a joint model with decentralized training data under the coordination of a centralized server. In contrast with centralized learning, the FL architecture allows for preserving clients' privacy and reducing the communication burden caused by transmitting data to the server. While there is a rich literature in distributed optimization in the context of machine learning, FL distinguishes itself from traditional distributed optimization in two key challenges: *high degrees of statistical and system heterogeneity* [13, 14, 18].

Limitations of Existing Methods. In an attempt to address the heterogeneity and improve the efficiency of FL, various optimization methods have been developed for FL. In particular, the federated averaging algorithm (`FedAvg`) [32] is the first state-of-the-art method for FL. In each communication round, `FedAvg` leverages local computation at each client and employs a centralized server to aggregate and update the global model parameter. While `FedAvg` has demonstrated empirical success in heterogeneous settings, it fails to fully address the underlying challenges associated with heterogeneity. For example, `FedAvg` randomly selects a subset of clients in each round regardless of their statistical heterogeneity, which diverge empirically in settings where data samples of each client follow a non-identical and independent distribution (non-IID).

Critical Learning Periods in FL. A recent trend of improving FL efficiency focuses on adaptive client selection during the FL training process, such as [5, 19, 25, 26, 38, 39, 43, 44, 47]. However, these studies implicitly assume that all learning phases during the FL training process are equally important. Unfortunately, this assumption has recently been revealed to be invalid due to the existence

KDD '23, August 6–10, 2023, Long Beach, CA

https://doi.org/10.1145/3580305.3599293

of *critical learning periods* (CLP), i.e., the final quality of a deep neural network (DNN) model is determined by the first few training epochs, in which deficits such as low quality or quantity of training data will cause irreversible model degradation, as illustrated in Figure 1(a). Notably, this phenomenon was revealed in the latest series of works in centralized learning [2, 8, 15, 16], and in FL [52, 54], which validated that if there is no sufficient training data at as early as the 20th communication round, the final test accuracy of FL is severely degraded compared to the standard FedAvg, as presented in Figure 1(b). Despite their insightful findings, there remains to be a major **gap** between the observation of CLP in FL and the goal of *more efficient training* and *improved model accuracy*, since existing client selection methods in state-of-the-art FL algorithms are *agnostic* of the existence of CLP in FL, which were only identified using a computationally expensive metric that emerges after the full training process in [52].

CLP Augmented Client Selection for Efficient FL. In this paper, we close this gap by demonstrating the importance of *augmenting* client selection with CLP in state-of-the-art FL algorithms. Through a range of carefully designed experiments on different machine learning models and datasets, we observe a consistently improved model accuracy without sacrificing communication efficiency by augmenting state-of-the-art FL algorithms with CLP. We build upon recent work by [52], who showed that if the training dataset for each client is not recovered to the entire training dataset early enough in the training process, the test accuracy of FL is permanently impaired (see Figure 1(b)). We extend this notation to client selection in FL and show that a larger number of clients are only required during the CLP. As a result, an adaptive and efficient client selection scheme is akin to finding CLP in the FL training process. These CLP can be detected in an online manner using a new metric called Federated Gradient Norm (FGN). To the best of our knowledge, this is the first step taken towards exploiting CLP for adaptive client selection for efficient FL to mitigate heterogeneity.

Main Contributions: We summarize our contributions:

- **Efficient Metric for CLP Detection.** We propose a practical, easy-to-compute Federated Gradient Norm (FGN) metric to identify CLP in an online manner, fixing a major paradox for connecting CLP with client selection for the efficient FL training goal.
- **Improved Model Accuracy and Communication Efficiency.** We propose a simple but powerful CLP augmented FL framework, dubbed as CriticalFL, that is generic across and orthogonal to different FL methods. In particular, we use FedAvg as our building block since it is the first and the most widely used one. CriticalFL inspects the changes in FGN to detect CLP in FL training process, and adaptively determines the number of clients to participate in each FL training round. With extensive empirical evaluation on different machine learning models with different datasets, we show that CriticalFL consistently achieves up to 9% accuracy improvement while maintaining better communication efficiency compared to FedAvg.
- **Generalization.** We show that CLP awareness can be easily combined with state-of-the-art FL methods, such as FedProx [26], VRL-SGD [28], FedNova [47], FedAdagrad, FedYogi, and FedAdam [36]. When augmented by CriticalFL via manipulating the client selection, existing methods achieve up to 8%, 9%, 9%, 10%, 11%, and 11% accuracy improvement, respectively, compared to training without being CLP augmented.

Algorithm 1 FedAvg

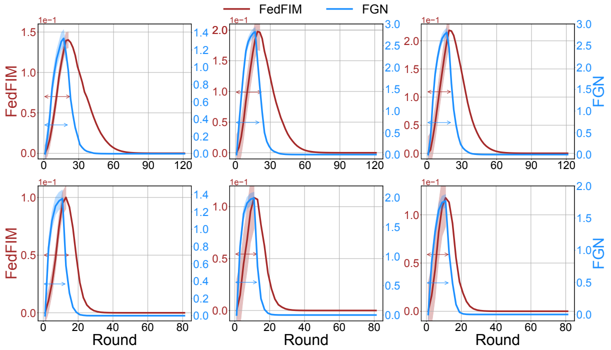
```
Input: $\mathcal{M}$, $\eta$, $E$, $\boldsymbol{\theta}^{(0)}$, $T$
for $t = 0, 1, \cdots, T-1$ do
    Server selects a subset $\mathcal{M}^{(t)}$ of $\mathcal{M}$ clients at random
    Server sends $\boldsymbol{\theta}^{(t)}$ to all selected clients
    Client $k \in \mathcal{M}^{(t)}$ updates $\boldsymbol{\theta}^{(t)}$ via $E$ iterations of SGD on $\mathcal{D}_k$ with stepsize $\eta$ to obtain $\boldsymbol{\theta}_k^{(t+1)}$
    Each selected client $k \in \mathcal{M}^{(t)}$ sends $\boldsymbol{\theta}_k^{(t+1)}$ back to the server
    Server aggregates the $\boldsymbol{\theta}$'s as $\boldsymbol{\theta}^{(t+1)} := \sum_{k \in \mathcal{M}^{(t)}} \frac{N_k}{\sum_{k \in \mathcal{M}^{(t)}} N_k} \boldsymbol{\theta}_k^{(t+1)}$
end for
```

2 RELATED WORK AND BACKGROUND

Critical Learning Periods (CLP). The presence of CLP in centralized neural network training was first highlighted in [2, 16]. Some other works [7, 8, 15, 17] have also highlighted the importance of early training phase in centralized learning. The existence of CLP in FL was recently discovered in [52]. In particular, they setup experiments where only partial datasets are available for the first few communication rounds and then continue training the model with entire training datasets for the rest of communication rounds. Surprisingly, the FL model trained in this way showed a permanent impaired test accuracy performance no matter how many additional training rounds are performed after CLP, as illustrated in Figure 1(b). However, studying CLP phenomena in FL [52] hinged on costly information metric (e.g., eigenvalues of the Hessian) that emerges after the full training, limiting their practical benefits. We differ from [52] by developing an easy-to-compute metric to identify CLP during the training process in an online manner.

Federated Optimization Setting. Consider the federated architecture where M clients jointly solve the optimization problem: $\min_{\boldsymbol{\theta} \in \mathbb{R}^d} F(\boldsymbol{\theta}) := \sum_{k=1}^{M} p_k F_k(\boldsymbol{\theta})$, where $p_k = N_k/N$ represents the relative sample size, and $F_k(\boldsymbol{\theta}) = \frac{1}{N_k} \sum_{\xi \in \mathcal{D}_k} \ell_k(\boldsymbol{\theta}; \xi)$ is the local objective function at the k-th client. Here ℓ_k denotes the loss function defined by the learning model, ξ represents a data sample from local dataset $\mathcal{D}_k$, and $\mathcal{M}$ denotes the set of clients.

Federated Learning and Client Selection. FedAvg [32] is the first to solve the above optimization problem through aggregating the locally trained models at the central server at the end of each communication round, and has sparked many follow-ups [5, 6, 12, 19–21, 26, 27, 31, 34, 36, 37, 39, 41–44, 47, 49, 51, 55]. For a comprehensive introduction to FL and other algorithmic variants in FL, we refer interested readers to [18]. Although the performance of FedAvg has been improved in both theory and practice by recent literature such as FedProx [26], FedNova [47], SCAFFOLD [19], VRL-SGD [28], FedBoost [10], FedMA [44], FetchSGD [38] and FedOPT [36], FedAvg is the first and the most widely used one. As



(a) $\alpha = 0.1$ (b) $\alpha = 0.2$ (c) $\alpha = 0.3$

Figure 2: Detecting CLP using FGN with $\delta = 0.01$ and FedFIM, where the shade and double-arrows indicate identified CLP. The results are conducted using AlexNet on (a) CIFAR-10 (top) and (b) Fashion-MNIST (bottom) datasets, which are non-IID partitioned across 128 clients using Dirichlet distributions $\text{Dir}_{128}(0.1)$, $\text{Dir}_{128}(0.2)$, and $\text{Dir}_{128}(0.3)$, respectively.

a result, we see FedAvg as our basic block. Specifically, at the initial step, the central server in FedAvg randomly initializes a global model $\boldsymbol{\theta}^{(0)}$. At each round, a *fixed* number of randomly selected clients run E iterations of local solver, e.g., the stochastic gradient descent (SGD) [45, 46, 56], and then the resulting model updates are averaged. The details of FedAvg are summarized in Algorithm 1, where $\mathcal{M}^{(t)} \subseteq \mathcal{M}$ and $m := |\mathcal{M}^{(t)}| \leq M, \forall t$.

Unlike most of aforementioned works that are agnostic to the existence of CLP, we design a novel CLP augmented FL framework. Importantly, we remark that *our proposed* CLP *augmented FL framework,* CriticalFL *is orthogonal to and can be easily combined with these methods (see Section 4)*, since CriticalFL *merely* augments a state-of-the-art FL method to adaptively determine the number of clients that participate in each FL training round, rather than changing the way how the FL method selects clients. Moreover, CriticalFL is also compatible with and complementary to other techniques such as gradient compression/quantization [4, 9].

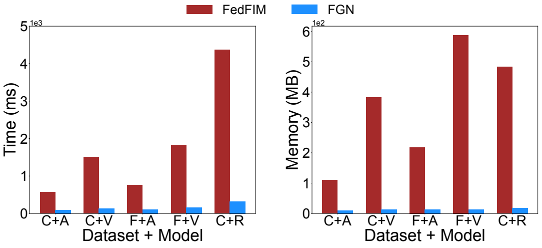
3 CRITICALFL FRAMEWORK

As motivated by aforementioned works, it is clear that finding an adaptive client selection scheme is akin to finding CLP in FL training process. To this end, we begin with how to efficiently detect CLP that lay out the rationale behind our framework. The rest of this section focuses on our proposed CriticalFL framework that augments client selection in state-of-the-art methods with CLP.

3.1 Detecting Critical Learning Periods

Prior works use the changes in eigenvalues of the Hessian or approximating the Hessian using (federated) Fisher information [2, 16, 52] as an indicator to detect CLP, which is computationally expensive (see Figure 3), and hence hard to be leveraged into client selection in an online manner. We deviate from these works and develop an approach based on federated gradient norm (FGN), which can be efficiently computed.

Specifically, we consider the difference in training loss for an individual data sample ξ and let $g(\boldsymbol{\theta}; \xi) = \frac{\partial}{\partial \boldsymbol{\theta}} \ell(\boldsymbol{\theta}; \xi)$ denote the gradient of the loss function evaluated on ξ. After performing a step SGD on this sample, the training loss $\Delta \ell = \ell(\boldsymbol{\theta} - \eta g(\boldsymbol{\theta}; \xi); \xi) - \ell(\boldsymbol{\theta}; \xi)$ can be approximated by its gradient norm using Taylor expansion, i.e., $\Delta \ell \approx -\eta \| g(\boldsymbol{\theta}; \xi) \|^2$. As a result, the overall training loss at the t-th round, which we define as the FGN, can be approximated using the weighted average of training loss across all selected clients, i.e.,



(a) Time (b) GPU

Figure 3: Computation time and memory consumption of FGN and FedFIM approaches to detect CLP.

$$\text{FGN}(t) = \sum_{k \in \mathcal{M}^{(t)}} \frac{N_k}{\sum_{k \in \mathcal{M}^{(t)}} N_k} \Delta \ell_k^{(t)}. \tag{1}$$

Then we develop a simple *threshold-based rule* to detect CLP based on FGN as follows: if

$$\frac{\text{FGN}(t) - \text{FGN}(t-1)}{\text{FGN}(t-1)} \geq \delta, \tag{2}$$

then round t is in CLP, where δ is the threshold used to declare CLP.

Experimental Validation. We compare the CLP identified by our FGN approach with the federated Fisher information (FedFIM) approach in [52]. From Figure 2, we observe that these two approaches yield similar results, but our FGN approach is much more computationally efficient (being orders of magnitude faster to compute). For example, the computation time and memory consumption of FGN and FedFIM under the same settings (in PyTorch [33] on Python 3 with three NVIDIA RTX A6000 GPUs, 48GB with 128GB RAM) are presented in Figure 3, where C+A, C+V, F+A, F+V and C+R represent AlexNet on CIFAR-10, VGG-11 on CIFAR-10, AlexNet on Fashion-MNIST, VGG-11 on Fashion-MNIST, and ResNet-18 on CIFAR-100, respectively. Hence our method can be easily leveraged for client selection during the training process in an online manner. More discussions on the robustness of our method can be found in [53].

3.2 The Design of CriticalFL Framework

We now describe CriticalFL, our proposed framework that adaptively determines the number of selected clients for FL training by leveraging identified CLP. Again, we use FedAvg as the building block, and our framework can be easily combined with other existing methods, which we will illustrate in Section 4.

Per our discussions on CLP, the final model accuracy is permanently impaired if not enough clients are involved in CLP no matter how much additional training is performed after CLP [52]. Thus, CriticalFL increases the number of selected clients of FedAvg from n_0 to $2n_0$, implying that more clients now participate in improving the global model in the next round during CLP. Using the model learned from the previous round $\boldsymbol{\theta}_{n_0}$ as the initial model, the $2n_0$ selected clients employ FedAvg and continue the learning procedure to reach a global model $\boldsymbol{\theta}_{2n_0}$. The procedure of geometrically increasing the number of selected clients continues till the set of selected clients contains all available M clients when the communication rounds are still in CLP (lines 5 and 8 in Algorithm 2).

Since more clients are selected during CLP in CriticalFL, this not only makes the comparison with FedAvg unfair, but also leads to much more communication between clients and the server. To this end, we leverage two insights to address these two concerns. On one hand, CriticalFL starts to gradually decrease the number of selected clients after CLP (line 12 in Algorithm 2), which is motivated by the fact that the final accuracy of using partial datasets is similar to that of using all datasets after CLP [52]. This not only makes the average number of selected clients in each round in CriticalFL comparable to that of FedAvg, but also improves the communication efficiency. On the other hand, the selected clients in CriticalFL only sends L parameters of its updated local model with the largest gradient derivations to the central server. For simplicity, the indicator of locations in the local updated parameter $\boldsymbol{\theta}_k$ of client k can be represented as $\boldsymbol{m}_k$, and hence only $\boldsymbol{\theta}_k \odot \boldsymbol{m}_k$ is shared with the server rather than $\boldsymbol{\theta}_k$ itself (line 6 in Algorithm 2). This is motivated by the observations that not all parameters are important in the training process [4, 9, 48], and sparsification method can be leveraged to further improve the communication efficiency of CriticalFL.

From a high-level perspective, CriticalFL exploits more clients in the initial phase of the learning procedure than a fixed number of clients for FedAvg in each round, to promptly reach a global model with higher accuracy since the initial learning phase plays a critical role in FL performance. By doing so, we hypothesize that the SGD is navigating to the steeper parts of the loss surface of the global model during CLP since a larger amount of data samples have contributed to the global model. However, the communication overhead of such an approach is relatively large since more clients are involved in FL training in each communication round. By only sharing top L local parameters of each client with the sever during CLP, and gradually decreasing the number of selected clients after CLP, the communication overhead of CriticalFL improves without hurting the final model accuracy. The key point is that more clients join the training process in the initial learning phase, and only a smaller number of clients is needed after CLP. As a result, CriticalFL consistently improves the model accuracy while maintains better communication efficiency than FedAvg.

Remark 1. *As our proposed* CriticalFL *provides a general framework to augment client selection with identified* CLP *in federated settings, one needs to specify the inner optimization subroutine (e.g., lines 2, 4, 7 and 11 in Algorithm 2) to quantify the improvement of the proposed approach. In particular, we set the subroutine to be* FedAvg *in Algorithm 2 since it is the most common algorithm and the building block of many variants in federated settings. This subroutine could be any federated learning algorithms (with possible variants), such as* FedProx *[26], VRL-SGD [28],* FedNova *[47],* FedAdagrad, FedYogi, *and* FedAdam *[36], which we will numerically illustrate in Section 4. In addition, each client in* CriticalFL *only sends top L parameters of its updated local model to the server during* CLP *(line 6 in Algorithm 2). However,* CriticalFL *is not limited to this, and can be easily generalized with other sparsification methods.*

Algorithm 2 CriticalFL: A CLP Augmented Client Selection Framework for Efficient Federated Learning

```
    Input: $\mathcal{M}, \eta, E, \boldsymbol{\theta}^{(0)}, T$
 1: for $t = 0, 1, \cdots, T-1$ do
 2:     Server selects a subset $\mathcal{M}^{(t)}$ of $\mathcal{M}$ clients at random
 3:     Server sends $\boldsymbol{\theta}^{(t)}$ to all selected clients
 4:     Client $k \in \mathcal{M}^{(t)}$ update the local model via FedAvg
 5:     if $\frac{\text{FGN}(t)-\text{FGN}(t-1)}{\text{FGN}(t-1)} \geq \delta$ then
 6:         Client $k \in \mathcal{M}^{(t)}$ sends $\boldsymbol{\theta}_k^{(t+1)} \odot \boldsymbol{m}_k^{(t+1)}$ to the server
 7:         Server aggregates the $\boldsymbol{\theta}$'s as:
            $\boldsymbol{\theta}^{(t+1)} := \sum_{k \in \mathcal{M}^{(t)}} \frac{N_k}{\sum_{k \in \mathcal{M}^{(t)}} N_k \boldsymbol{m}_k^{(t+1)}} \boldsymbol{\theta}_k^{(t+1)} \odot \boldsymbol{m}_k^{(t+1)}$
 8:         $|\mathcal{M}^{(t+1)}| \leftarrow \min\{2|\mathcal{M}^{(t)}|, M\}$ //Double clients in CLP
 9:     else
10:         Client $k \in \mathcal{M}^{(t)}$ sends $\boldsymbol{\theta}_k^{(t+1)}$ to the server
11:         Server aggregates local models via FedAvg
12:         $|\mathcal{M}^{(t+1)}| \leftarrow \max\{\frac{1}{2}|\mathcal{M}^{(t)}|, \frac{1}{2}m\}$ //Halve clients after CLP
13:     end if
14: end for
```

4 EXPERIMENTS

In this section, we evaluate the performance of our CriticalFL framework. Our results address the following questions:

- What is the benefit of using our CriticalFL framework compared to FedAvg in terms of final test accuracy and communication efficiency (Section 4.2)?
- What is the generalization performance of our CriticalFL framework when its inner optimization subroutine (e.g., lines 2, 4, 7 and 11 in Algorithm 2) is replaced by other state of the arts (Section 4.3)?
- How do different hyperparameters impact the performance of our CriticalFL framework (Section 4.4)?

For sake of readability, some experimental results and details are relegated to [53].

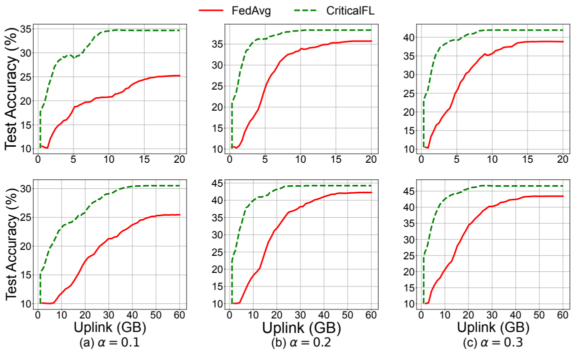
4.1 Experiment Setup

We consider two tasks: (i) image classification using CIFAR-10 and CIFAR-100 [23], and Fashion-MNIST [50] datasets; and (ii) next-character prediction on the dataset of *The Complete Works of William Shakespeare* (Shakespeare) [32]. We use four representative DNN models: AlexNet [24] and VGG-11 [40] for CIFAR-10 and Fashion-MNIST, ResNet-18 [11] for CIFAR-100, and a stacked character-level LSTM language model as in [22, 32] for Shakespeare.

We simulate the non-IID FL scenario by considering a heterogeneous partition for which the number of data points and class proportions are unbalanced. In particular, we simulate a heterogeneous partition into M clients by sampling $\boldsymbol{p}_k \sim \text{Dir}_M(\alpha)$, where α is the parameter of the Dirichlet distribution. We choose $\alpha = 0.1, 0.2, 0.3$ in our experiments as done in [44, 47]. The level of heterogeneity among local datasets across different clients can be reduced when α increases. We consider the total number of clients to be 128. The local learning rate η is initialized as 0.01 and decayed by a constant factor after each communication round. We set the weight decay to be 10^{-5}. The detection threshold is $\delta = 0.01$ and the number of local

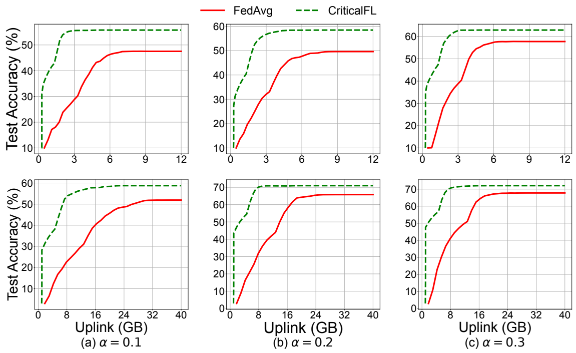
Dataset (Model)	Non-IID Degree	FedAvg		FedProx		VRL-SGD		FedNova		FedAdagrad		FedYogi		FedAdam	
		Original	CLP	Original	CLP	Original	CLP	Original	CLP	Original	CLP	Original	CLP	Original	CLP
CIFAR-10 (AlexNet)	$\alpha = 0.1$	25.25±0.5	**34.66**±0.1	27.19±0.5	**32.81**±0.5	26.45±0.5	**35.65**±0.5	25.92±0.5	**32.81**±0.5	26.01±0.5	**35.98**±0.1	27.03±0.2	**36.42**±0.5	27.39±0.2	**36.84**±0.5
	$\alpha = 0.2$	35.74±0.3	**38.33**±0.5	35.26±0.5	**37.85**±0.5	35.66±0.5	**38.36**±0.5	35.91±0.5	**38.45**±0.4	36.34±0.4	**41.79**±0.1	36.44±0.1	**41.55**±0.2	37.08±0.3	**42.37**±0.5
	$\alpha = 0.3$	38.82±0.1	**41.94**±0.1	36.72±0.5	**40.82**±0.5	39.78±0.5	**41.93**±0.5	38.75±0.5	**41.88**±0.5	40.61±0.2	**43.97**±0.1	40.58±0.1	**44.98**±0.1	40.19±0.3	**44.35**±0.4
CIFAR-10 (VGG-11)	$\alpha = 0.1$	25.43±0.1	**30.53**±0.2	25.97±0.5	**31.55**±0.5	26.74±0.5	**30.79**±0.5	25.53±0.5	**28.78**±0.5	24.75±0.1	**32.46**±0.2	26.27±0.2	**34.30**±0.1	27.38±0.3	**34.31**±0.5
	$\alpha = 0.2$	42.26±0.1	**44.22**±0.2	40.41±0.5	**42.47**±0.2	41.56±0.4	**44.65**±0.2	41.03±0.5	**43.46**±0.1	43.35±0.1	**47.52**±0.1	43.87±0.1	**49.24**±0.1	45.73±0.2	**51.05**±0.1
	$\alpha = 0.3$	43.43±0.1	**46.61**±0.1	41.12±0.4	**46.32**±0.2	43.67±0.1	**46.58**±0.1	43.86±0.3	**48.13**±0.1	45.62±0.1	**50.16**±0.1	45.17±0.1	**51.03**±0.1	46.99±0.2	**52.67**±0.1
Fashion MNIST (AlexNet)	$\alpha = 0.1$	47.53±0.5	**55.77**±0.2	47.53±0.5	**55.31**±0.5	49.08±0.5	**57.14**±0.5	49.45±0.5	**56.23**±0.2	48.68±0.5	**57.98**±0.2	47.88±0.5	**58.12**±0.2	48.11±0.3	**58.62**±0.1
	$\alpha = 0.2$	49.61±0.5	**58.45**±0.1	50.48±0.5	**58.27**±0.5	49.12±0.5	**57.44**±0.3	49.48±0.5	**58.27**±0.5	49.42±0.2	**59.52**±0.1	50.76±0.3	**59.39**±0.5	50.68±0.5	**60.73**±0.1
	$\alpha = 0.3$	57.75±0.4	**62.90**±0.1	54.41±0.5	**61.11**±0.2	56.54±0.5	**62.67**±0.2	55.80±0.5	**62.42**±0.3	60.27±0.2	**66.31**±0.1	60.55±0.4	**67.06**±0.1	62.07±0.3	**66.23**±0.1
Fashion MNIST (VGG-11)	$\alpha = 0.1$	51.92±0.5	**58.76**±0.2	52.36±0.5	**63.02**±0.5	53.28±0.5	**64.18**±0.5	50.67±0.5	**62.91**±0.5	55.92±0.5	**64.27**±0.2	55.32±0.1	**64.09**±0.5	55.67±0.5	**65.86**±0.1
	$\alpha = 0.2$	65.79±0.2	**70.96**±0.1	65.16±0.5	**69.50**±0.2	66.95±0.5	**70.62**±0.1	67.67±0.1	**70.23**±0.1	69.33±0.1	**74.52**±0.1	71.03±0.1	**75.25**±0.1	70.68±0.2	**75.89**±0.1
	$\alpha = 0.3$	67.77±0.2	**72.04**±0.1	66.53±0.3	**71.91**±0.2	68.27±0.4	**73.20**±0.2	68.26±0.2	**72.59**±0.1	70.89±0.2	**74.32**±0.1	72.39±0.2	**75.63**±0.1	72.81±0.2	**75.91**±0.1
CIFAR-100 (ResNet-18)	$\alpha = 0.1$	28.28±0.5	**31.18**±0.5	27.83±0.5	**31.44**±0.2	28.57±0.5	**32.02**±0.3	28.76±0.5	**31.79**±0.2	29.21±0.5	**33.01**±0.3	28.94±0.5	**33.38**±0.4	28.57±0.5	**32.50**±0.4
	$\alpha = 0.2$	30.71±0.4	**32.54**±0.4	30.65±0.5	**32.44**±0.2	30.76±0.5	**32.82**±0.2	30.82±0.5	**33.05**±0.2	32.37±0.5	**34.35**±0.3	32.18±0.5	**34.78**±0.4	31.85±0.5	**34.65**±0.3
	$\alpha = 0.3$	31.76±0.5	**32.95**±0.3	31.77±0.3	**33.19**±0.1	31.58±0.3	**32.96**±0.2	31.88±0.3	**33.29**±0.2	32.78±0.5	**35.56**±0.4	32.91±0.5	**35.74**±0.3	32.58±0.5	**35.06**±0.2
Shakespeare (LSTM)	$\alpha = 0.1$	40.92±0.4	**44.06**±0.3	40.80±0.5	**43.35**±0.5	40.89±0.5	**44.12**±0.5	40.98±0.5	**44.22**±0.4	41.39±0.3	**44.89**±0.4	41.23±0.4	**44.32**±0.3	41.45±0.4	**44.36**±0.3
	$\alpha = 0.2$	43.90±0.5	**46.76**±0.2	43.73±0.5	**46.30**±0.4	43.98±0.5	**46.94**±0.4	43.93±0.4	**46.85**±0.3	44.45±0.2	**46.91**±0.1	44.56±0.2	**46.77**±0.2	44.55±0.2	**46.63**±0.2
	$\alpha = 0.3$	45.05±0.3	**47.62**±0.1	44.86±0.3	**47.42**±0.2	45.02±0.3	**47.59**±0.2	45.04±0.2	**47.79**±0.1	45.37±0.1	**48.25**±0.4	45.42±0.1	**47.91**±0.1	45.35±0.2	**47.82**±0.1

Table 1: Final test accuracy of state-of-the-art FL algorithms (the “Original” columns), and the corresponding CLP augmented method (the “CLP” columns) via our CriticalFL framework using various non-IID partitioned datasets with different models.



(a) $\alpha = 0.1$ (b) $\alpha = 0.2$ (c) $\alpha = 0.3$

Figure 4: Test accuracy of FedAvg and CriticalFL using (top) AlexNet and (bottom) VGG-11 on non-IID CIFAR-10.



(a) $\alpha = 0.1$ (b) $\alpha = 0.2$ (c) $\alpha = 0.3$

Figure 5: Test accuracy of FedAvg and CriticalFL using (top) AlexNet and (bottom) VGG-11 on non-IID Fashion-MNIST.

training epochs is $E = 2$. We choose $L = 20\%$. An ablation study is conducted in Section 4.4 to investigate the impact of these hyperparameters. We implement CriticalFL and considered baselines in PyTorch [33] on Python 3 with three NVIDIA RTX A6000 GPUs. We run each experiment with three independent trials and report the mean results. For ease of presentation, we omit the variances which are observed to be small in the experiments.

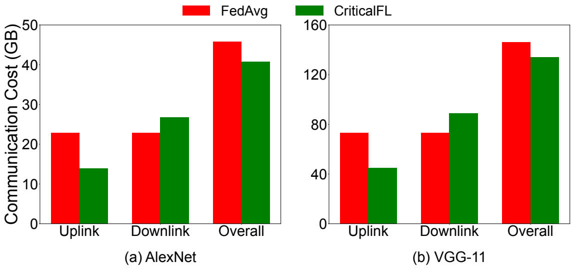
4.2 Importance of being CLP Augmented: CriticalFL vs. FedAvg

In this experiment, we study the performance of CriticalFL with accuracy and communication efficiency. Our goal is to compare CriticalFL to FedAvg in terms of the final test accuracy and the communication efficiency, including the communication costs measured by the amount of data (i.e., model parameters) transitions between clients and the central server to achieve the final test accuracy, and the number of communication rounds needed for the global model to achieve good performance on the test data.

Test Accuracy. The final test accuracy of CriticalFL and FedAvg on non-IID partitioned datasets with FedAvg selecting 16 clients in each round are summarized in Table 1 (the two columns corresponding to FedAvg). For ease of readability, we only present the test accuracy over uplink communication costs on CIFAR-10 and Fashion-MNIST in shown in Figures 4 and 5, respectively.

Obviously, CriticalFL consistently outperforms FedAvg in all scenarios with an improved final test accuracy up to 9%. Its advantage is especially pronounced when the dataset is partitioned across clients using a Dirichlet distribution with parameter 0.1, i.e., the datasets across clients are highly non-IID. Not surprisingly, we observe the importance of being CLP augmented in training efficiency, which is fully reflected via the test accuracy. For example in Figures 4(a) and 5(a), CriticalFL exhibits a dramatic accuracy increase in the early phase of the FL training process. This coincides with the fact that CriticalFL selects a larger number of clients in each round in the early phase due to being CLP augmented (lines 5-8 in Algorithm 2). Though the accuracy slightly decreases in a short period due to the decreased number of selected clients (lines 10-12 in Algorithm 2), the final test accuracy is significantly improved. Our findings on the importance of being CLP augmented in the FL training process, e.g., for client selection, seem to be consistent with recently reported observations that the initial learning phase plays a key role in determining the outcome of the training process.

Communication Efficiency. The benefit of being CLP augmented for FL training is further reflected via communication efficiency. In



(a) AlexNet (b) VGG-11

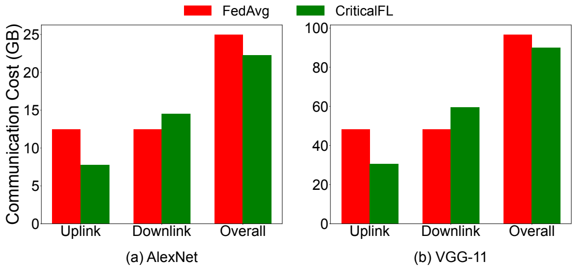
Figure 6: Communication costs of FedAvg and CriticalFL to achieve the final test accuracy reported in Table 1 on non-IID CIFAR-10.

Dataset (Model)	Non-IID Degree	FedAvg	CriticalFL
CIFAR-10 (AlexNet)	$\alpha = 0.1$	83	**29**
	$\alpha = 0.2$	77	**41**
	$\alpha = 0.3$	71	**43**
CIFAR-10 (VGG-11)	$\alpha = 0.1$	77	**57**
	$\alpha = 0.2$	75	**55**
	$\alpha = 0.3$	71	**43**
Fashion MNIST (AlexNet)	$\alpha = 0.1$	43	**25**
	$\alpha = 0.2$	49	**27**
	$\alpha = 0.3$	39	**26**
Fashion MNIST (VGG-11)	$\alpha = 0.1$	49	**27**
	$\alpha = 0.2$	39	**25**
	$\alpha = 0.3$	35	**24**

Table 2: Communication rounds required by FedAvg and CriticalFL to achieve a targeted accuracy on non-IID CIFAR-10 and Fashion-MNIST.

particular, we consider the communication costs for CriticalFL and FedAvg to achieve the final test accuracy reported in Table 1, as well as the communication rounds required by CriticalFL and FedAvg to achieve a targeted test accuracy.

In FL settings, the central server sends parameters to selected clients via the downlink that connects the server and clients, while clients send updated local model to the server via the corresponding uplink. We report the corresponding downlink, uplink and overall communication costs of CriticalFL and FedAvg to achieve the final test accuracy reported in Table 1 on non-IID CIFAR-10 and Fashion-MNIST in Figures 6 and 7, respectively. On one hand, we observe that CriticalFL significantly reduces the uplink communication costs compared to that of FedAvg. This is due to two intrinsic design properties in CriticalFL: (i) the selected clients during CLP only share a subset of parameters of their local updated models with the server (line 6 in Algorithm 2); and (ii) more clients are only needed during CLP, and a smaller number of clients are needed afterwards (lines 5, 8, and 12 in Algorithm 2), in particular, the average number of clients involved in each round in CriticalFL is 0.95× to 1.02× that of FedAvg for achieving the final test accuracy. On the other hand, we observe that the downlink communication costs of CriticalFL is higher than that of FedAvg. This is because the server sends the full model to selected clients and more clients are involved in the early phases in CriticalFL. Importantly, in real-world systems, the bandwidth of uplinks often imposes a tighter bottleneck than that of downlink, e.g., the average uplink bandwidth is less than one fourth of downlink bandwidth [1, 30, 35].



(a) AlexNet (b) VGG-11

Figure 7: Communication costs of FedAvg and CriticalFL to achieve the final test accuracy reported in Table 1 on non-IID Fashion-MNIST.

Hence, CriticalFL brings benefits on communication costs compared to FedAvg, especially on the more constrained uplink. More importantly, CriticalFL reduces the overall communication costs compared to that of FedAvg by up to 12%.

We further report the communication rounds required by FedAvg and CriticalFL to achieve any targeted accuracy. Since the final test accuracy of CriticalFL is higher than that of FedAvg, we set the targeted accuracy to be the final test accuracy of FedAvg as reported in Table 1 (or Figures 4 and 5, respectively). Comparisons on other targeted accuracy can be found in [53]. It is clear from Table 2 that CriticalFL requires fewer rounds to achieve the same test accuracy. Again this advantage is pronounced on highly non-IID data partitions.

4.3 Generalization

As mentioned earlier, our proposed CLP augmented FL framework, CriticalFL is orthogonal to existing state-of-the-art methods, and hence can be easily combined with these methods by simply replacing the inner optimization subroutine (FedAvg) in Algorithm 2 with the corresponding methods. To this end, we study the generalization of CriticalFL and consider six state of the arts, i.e., FedProx [26], VRL-SGD [28], FedNova [47], as well as FedOPT [36] with three methods, i.e., FedAdagrad, FedYogi and FedAdam. We call the corresponding CLP augmented methods as CriticalProx, CriticalVRL, CriticalNova, CriticalAdagrad, CriticalYogi and CriticalAdam, respectively. We notice that the performance of FedProx depends on the hyperparameter μ, i.e., the coefficient associated with the proximal term of each local objective. We tune this parameter using grid search and report the best value of $\mu = 0.01$ for AlexNet experiments and $\mu = 0.001$ for all other models.

In Table 1, we present the final test accuracy on non-IID datasets across 128 clients with FedProx, VRL-SGD, FedNova, FedAdagrad, FedYogi and FedAdam selecting 16 clients in each round. Due to space constraints, we only report the test accuracy over communication rounds on CIFAR-10 in Figure 8. Again, CLP augmented significantly improves the performance of existing methods, i.e., CriticalProx, CriticalVRL, CriticalNova, CriticalAdagrad, CriticalYogi and CriticalAdam outperform FedProx, VRL-SGD, FedNova, FedAdagrad, FedYogi and FedAdam, respectively, in all scenarios with an improved final test accuracy up to 9%, 9%, 9%, 10%, 11% and 11%, respectively, as shown in Table 1 and Figure 8, respectively. Its advantage is especially pronounced on highly non-IID dataset across clients in Figure 8(a). Likewise, the CLP augmented

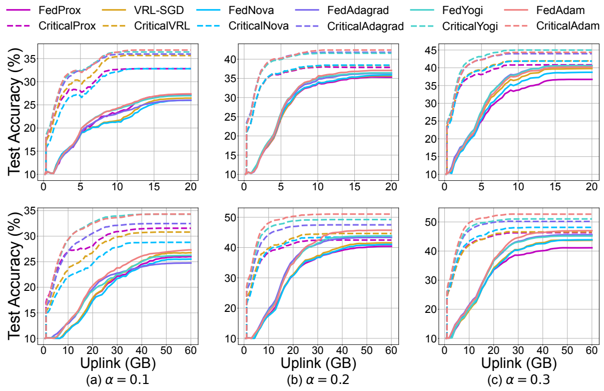


Figure 8: Test accuracy of FedProx, VRL-SGD, FedNova, FedAdagrad, FedYogi and FedAdam, as well as CriticalProx, CriticalVRL, CriticalNova, CriticalAdagrad, CriticalYogi and CriticalAdam, using (top) AlexNet and (bottom) VGG-11 on non-IID CIFAR-10.

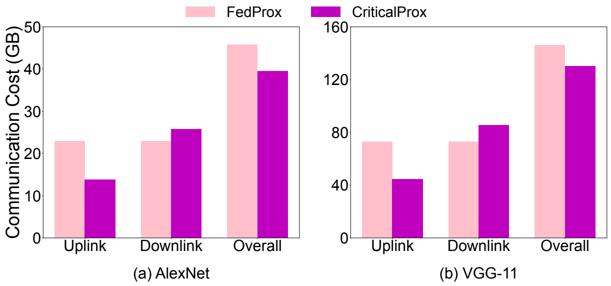


Figure 9: Communication costs of FedProx and CriticalProx on non-IID CIFAR-10.

Dataset (Model)	Non-IID Degree	FedProx	CriticalProx
CIFAR-10 (AlexNet)	$\alpha = 0.1$	77	**37**
	$\alpha = 0.2$	75	**40**
	$\alpha = 0.3$	73	**36**
CIFAR-10 (VGG-11)	$\alpha = 0.1$	75	**35**
	$\alpha = 0.2$	76	**43**
	$\alpha = 0.3$	71	**36**

Table 3: Communication rounds required by FedProx and CriticalProx to achieve a targeted accuracy on non-IID CIFAR-10.

method, e.g., CriticalProx leads to a smaller overall communication cost to achieve the final test accuracy compared to that of the corresponding baseline FedProx, as shown in Figure 9, while maintaining a comparable average number of clients involved in each round. Similarly, CriticalProx requires fewer communication rounds to achieve a targeted accuracy, which is chosen to be the final test accuracy of FedProx than FedProx itself, as shown in Table 3. Similar observations can be made for other five comparisons, which can be found in [53].

4.4 Ablation Study

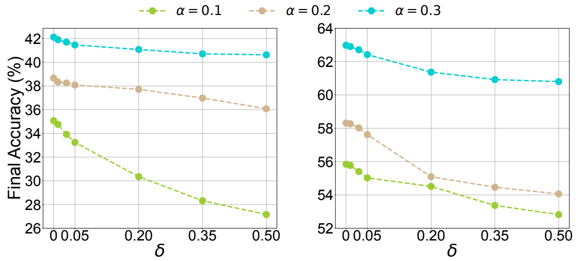
In this subsection, we conduct a comprehensive ablation study to investigate the impacts of different hyperparameters in the design of our CriticalFL framework. For ease of readability, we only present experimental results with FedAvg, FedProx, VRL-SGD and FedNova. Similar results hold for FedAdagrad, FedYogi and FedAdam, and hence are omitted here.

Detection Thresholds. As discussed in Figure 2, our experiments reveal that CLP can be efficiently identified using FGN via a simple threshold-type rule in Equation (2). We now evaluate the sensitivity of the threshold value δ used to declare CLP. The candidate values are $\{0, 0.01, 0.03, 0.05, 0.2, 0.35, 0.5\}$, and the corresponding final test accuracy of CriticalFL using AlexNet on non-IID CIFAR-10 and Fashion-MNIST is illustrated in Figure 10. When data partitions are highly non-IID (i.e., $\alpha = 0.1$), the CLP declaration determined by δ has an observable effect on the final accuracy. This is because as δ becomes larger, fewer rounds in the initial phase are declared as CLP by Equation (2). As a result, the effect of being CLP augmented on the final test accuracy is shallowed since CriticalFL only uses a larger number of clients in fewer rounds compared to FedAvg according to Algorithm 2. On the other hand, CriticalFL is robust to the detection process, i.e., tolerant to detection errors with different threshold values when data partitions are not highly non-IID. Similar observations can be made for CriticalProx, CriticalVRL, CriticalNova, CriticalAdagrad, CriticalYogi and CriticalAdam and hence are relegated to [53]. Thus we set $\delta = 0.01$ in our experiments.

Non-IID Degree. We simulate a heterogeneous data partition into M clients using the Dirichlet distribution with parameter α. From Figure 11, we observe that being CLP augmented consistently improves the final test accuracy of a state-of-the-art method across all values of α in consideration. For example, CriticalFL always outperforms FedAvg, and CriticalProx always outperforms FedProx. The benefits of being CLP augmented are especially pronounced when the datasets across clients are highly non-IID (i.e., a smaller value of α). Hence, we choose $\alpha = 0.1, 0.2, 0.3$ for illustrations in above experiments. For ease of readability, we set $\alpha = 0.1$ in the rest of ablation studies. Similar observations can be made with $\alpha = 0.2, 0.3$ and hence are omitted. In addition, as the non-IID degree decreases (as α increases), the final test accuracy of CriticalFL, CriticalProx, CriticalVRL, CriticalNova, CriticalAdagrad, CriticalYogi and CriticalAdam increases. This is consistent with recently reported observations, e.g., in [3, 29] that non-IID degree degrades the model final test accuracy.

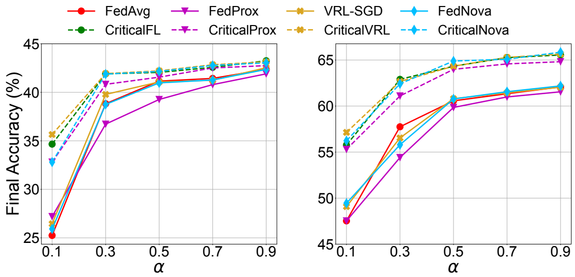
Local Training Epochs. We note that the number of local training epochs (denoted E) is a common parameter shared by considered baselines, which reportedly has an impact on the performance of FedAvg [32, 44]. To this end, we evaluate the effect of E using AlexNet on non-IID CIFAR-10 and Fashion-MNIST with $\alpha = 0.1$. The candidate local epochs we consider are $E \in \{1, 2, 3, 4, 5\}$ as done in [47]. From Figure 12, we observe that increasing the number of local epochs improves the final test accuracy in general, and being CLP augmented consistently improves the final test accuracy of state-of-the-art methods across all values of E. Since the gains in test accuracy exhibit the "diminishing return effect" as the number of local epochs increases, we set $E = 2$ in our experiments.

Weight Decay. Though the CLP in FL are robust to weight decays as reported in [52], the final test accuracy using AlexNet on non-IID CIFAR-10 and Fashion-MNIST with $\alpha = 0.1$ is still affected with weight decays as shown in Figure 13. Again, we consistently



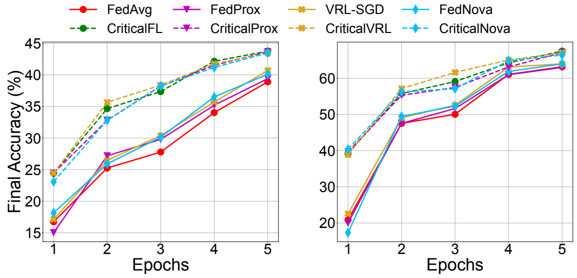
(a) CIFAR-10 (b) Fashion MNIST

Figure 10: Sensitivity of detection threshold.



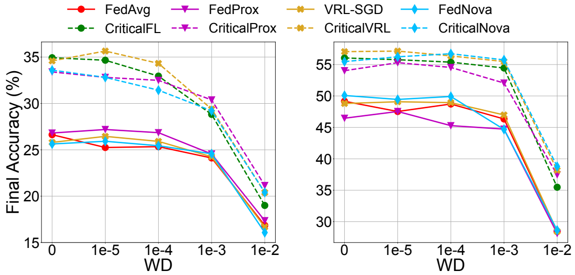
(a) CIFAR-10 (b) Fashion MNIST

Figure 11: Impact of non-IID degree.



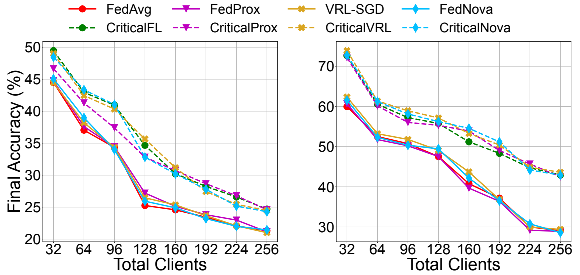
(a) CIFAR-10 (b) Fashion MNIST

Figure 12: Effect of local training epochs.



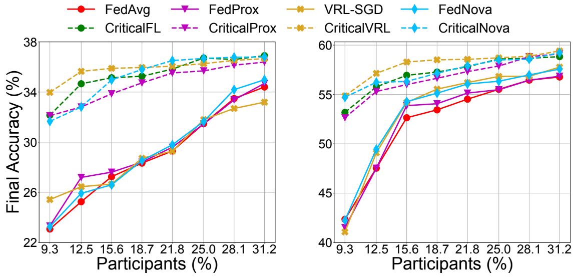
(a) CIFAR-10 (b) Fashion MNIST

Figure 13: Effect of weight decays.



(a) CIFAR-10 (b) Fashion MNIST

Figure 14: Impact of number of clients.



(a) CIFAR-10 (b) Fashion MNIST

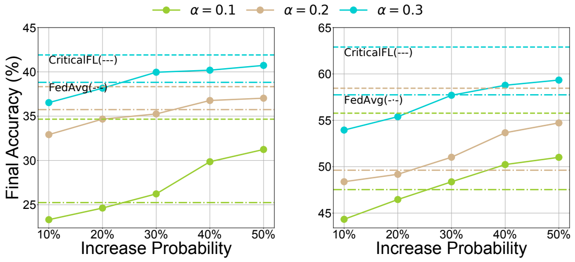
Figure 15: Impact of client participation rate.

observe the benefits of being CLP augmented across all values of weight decays. Since the advantage decreases as the weight decay increases, we set the weight decay to be 10^{-5} in our experiments.

Number of Clients. In all of our above experiments, we consider a FL setting with 128 clients in total. We now consider the same experimental settings as above besides varying the total number of clients in the system using AlexNet on non-IID CIFAR-10 and Fashion-MNIST with $\alpha = 0.1$. As shown in Figure 14, the advantage of being CLP augmented exhibits across all settings, i.e., CriticalFL (resp. CriticalProx) outperforms FedAvg (resp. FedProx) regardless of the total number of clients. Without loss of generality, we choose $M = 128$ in our experiments.

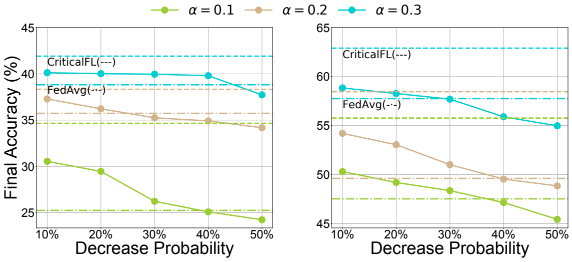
Client Participation Rate. In all of our experiments, our considered baselines select 16 out of 128 clients to participate in each training round, i.e., the participation rate is 12.5%. We now investigate the impact of client participation rates on the final test accuracy and being CLP augmented using AlexNet on non-IID CIFAR-10 and Fashion-MNIST with $\alpha = 0.1$. Again, when a state-of-the-art method is augmented with the CLP, the final test accuracy is consistently improved across all client participation rates as shown in Figure 15. The advantage is particularly pronounced with a low participation rate. This is quite intuitive since in our CLP augmented framework, CriticalFL selects more clients during CLP than FedAvg (see line 8 in Algorithm 2), and hence the benefits are more obvious when FedAvg has a low client participation rate. We select 16 clients, i.e., a 12.5% participation rate for all state-of-the-art methods via considering the tradeoff between final test accuracy and benefits of being CLP augmented.

Randomly Increasing and Decreasing the Number of Selected Clients. Besides deterministically increasing or decreasing the number of selected clients as in CriticalFL, we randomly increase or decrease the number. Specifically, we consider two settings. On one hand, we fix the probability of decreasing the selected clients from m to m/2 to be 30% in each round, and investigate the impact of the probability of increasing the selected clients from m to 2m in each round. On the other hand, we fix the probability



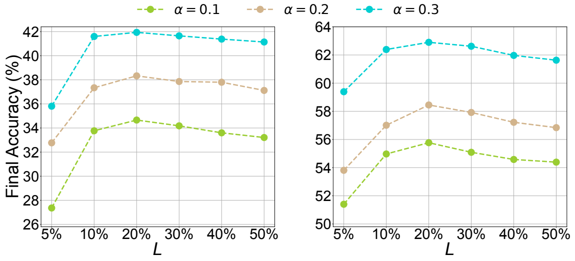
(a) CIFAR-10 (b) Fashion MNIST

Figure 16: Test accuracy of `CriticalFL` when randomly increase the number of participated clients.



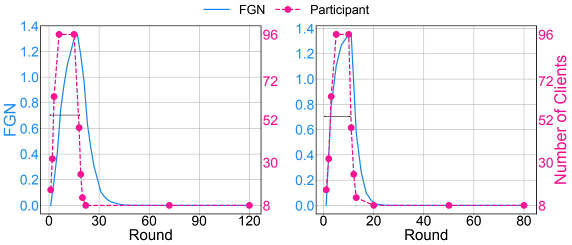
(a) CIFAR-10 (b) Fashion MNIST

Figure 17: Test accuracy of `CriticalFL` when randomly decrease the number of participated clients.



(a) CIFAR-10 (b) Fashion MNIST

Figure 18: Impact of the number of local parameters transmitted during `CLP` on the final test accuracy of `CriticalFL`.



(a) CIFAR-10 (b) Fashion MNIST

Figure 19: Relations between FGN and the number of selected clients in each round.

of increasing the selected clients from m to 2m to be 30% in each round, and investigate the impact of the probability of decreasing the selected clients from m to m/2 in each round. We report the final test accuracy of the model, and compare it with `CriticalFL` (see Algorithm 2) and `FedAvg`. As shown in Figures 16 and 17, random increase or decrease may not necessarily improve the performance of `FedAvg`. This is due to the fact that the random strategy may not necessarily align with the findings of `CLP` [2, 52] that more data/clients need to be involved in early training phases.

Number of Local Parameters. To improve the communication efficiency, all selected clients during `CLP` only share top L local parameters with the server in our `CriticalFL` (line6 in Algorithm 2). We set the $L = 20\%$ in our experiments and now evaluate its impact. The final test accuracy of `CriticalFL` with different values of L is shown in Figure 18. On one hand, if L is too small, then not enough parameters (i.e., updated model information) is transmitted to the server, and hence will degrade the final test accuracy. On the other hand, when L is larger, some parameters are not that much important, and hence brings marginal improvement at the cost of communications. Similar observations can be made in other baseline methods. Hence, we set $L = 20\%$ in our experiments.

Relations between the Number of Selected Clients and the FGN. We report the number of selected clients and the FGN curve in Figure 19. We observe that more clients are involved in the early training phases where `CLP` occur. This is consistent with the design of our `CriticalFL` framework, where more clients are only needed in the initial learning phase (see Section 3.2).

5 CONCLUSION

In this paper, we presented `CriticalFL`, a simple but powerful `CLP` augmented FL framework for adaptive client selection. `CriticalFL` worked by adaptively choosing more clients in `CLP` during the FL training process and fewer clients elsewhere. We proposed a practical and easy-to-compute federated gradient norm (FGN) metric to identify such `CLP` during the training process in an online manner. We showed that `CriticalFL` significantly improved the final test accuracy by up to 11% compared to its counterpart `FedAvg` using different models and datasets, while maintaining comparable or even better communication efficiency. Finally, we illustrated the generalization of our proposed `CLP` augmented framework via manipulating the client selection of state-of-the-art methods augmented by `CriticalFL`. In the future work, we want to extend `CriticalFL` to improve FL of different machine learning models on other popular techniques such as gradient compression/quantization, fair aggregation, personalization, and adversarial attacks. We also believe that it is important to study the performance of `CriticalFL` on other models and datasets.

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation (NSF) grants 2104880, 2148309, 2153502, 1763620, 1948374, and 2146447, and U.S. Army Research Office (ARO) grant W911NF-23-1-0072. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agencies.

REFERENCES

[1] Reisizadeh A., Mokhtari A., and Hassani H. 2020. Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization. In *Proc. of AISTATS.*

[2] Alessandro Achille, Matteo Rovere, and Stefano Soatto. 2019. Critical Learning Periods in Deep Networks. In *Proc. of ICLR.*

[3] Idan Achituve, Aviv Shamsian, Aviv Navon, Gal Chechik, and Ethan Fetaya. 2021. Personalized Federated Learning with Gaussian Processes. *Proc. of NeurIPS* (2021).

[4] Debraj Basu, Deepesh Data, Can Karakus, and Suhas Diggavi. 2019. Qsparse-local-SGD: Distributed SGD with Quantization, Sparsification and Local Computations. In *Proc. of NeurIPS.*

[5] 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).

[6] Yae Jee Cho, Jianyu Wang, and Gauri Joshi. 2022. Towards Understanding Biased Client Selection in Federated Learning. In *Proc. of AISTATS.*

[7] Jonathan Frankle, David J Schwab, and Ari S Morcos. 2020. The Early Phase of Neural Network Training. In *Proc. of ICLR.*

[8] Aditya Sharad Golatkar, Alessandro Achille, and Stefano Soatto. 2019. Time Matters in Regularizing Deep Networks: Weight Decay and Data Augmentation Affect Early Learning Dynamics, Matter Little Near Convergence. *Proc. of NeurIPS* (2019).

[9] Farzin Haddadpour, Mohammad Mahdi Kamani, Aryan Mokhtari, and Mehrdad Mahdavi. 2021. Federated Learning with Compression: Unified Analysis and Sharp Guarantees. In *Proc. of AISTATS.*

[10] Jenny Hamer, Mehryar Mohri, and Ananda Theertha Suresh. 2020. FedBoost: A Communication-Efficient Algorithm for Federated Learning. In *Proc. of ICML.*

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *Proc. of IEEE CVPR.*

[12] Samuel Horváth and Peter Richtarik. 2021. A Better Alternative to Error Feedback for Communication-Efficient Distributed Learning. In *Proc. of ICLR.*

[13] Ahmed Imteaj, Khandaker Mamun Ahmed, Urmish Thakker, Shiqiang Wang, Jian Li, and M Hadi Amini. 2022. Federated Learning for Resource-Constrained IoT Devices: Panoramas and State of the Art. *Federated and Transfer Learning* (2022), 7–27.

[14] Ahmed Imteaj, Urmish Thakker, Shiqiang Wang, Jian Li, and M Hadi Amini. 2021. A survey on federated learning for resource-constrained IoT devices. *IEEE Internet of Things Journal* 9, 1 (2021), 1–24.

[15] Stanislaw Jastrzebski, Devansh Arpit, Oliver Astrand, Giancarlo B Kerg, Huan Wang, Caiming Xiong, Richard Socher, Kyunghyun Cho, and Krzysztof J Geras. 2021. Catastrophic Fisher Explosion: Early Phase Fisher Matrix Impacts Generalization. In *Proc. of ICML.*

[16] Stanislaw Jastrzebski, Zachary Kenton, Nicolas Ballas, Asja Fischer, Yoshua Bengio, and Amos J Storkey. 2019. On the Relation Between the Sharpest Directions of DNN Loss and the SGD Step Length. In *Proc. of ICLR.*

[17] Stanislaw Jastrzebski, Maciej Szymczak, Stanislav Fort, Devansh Arpit, Jacek Tabor, Kyunghyun Cho, and Krzysztof Geras. 2020. The Break-Even Point on Optimization Trajectories of Deep Neural Networks. In *Proc. of ICLR.*

[18] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. 2019. Advances and Open Problems in Federated Learning. *arXiv preprint arXiv:1912.04977* (2019).

[19] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. 2020. SCAFFOLD: Stochastic Controlled Averaging for Federated Learning. In *Proc. of ICML.*

[20] Angelos Katharopoulos and François Fleuret. 2018. Not All Samples Are Created Equal: Deep Learning with Importance Sampling. In *Proc. of ICML.*

[21] Ahmed Khaled, Konstantin Mishchenko, and Peter Richtárik. 2020. Tighter theory for local SGD on identical and heterogeneous data. In *Proc. of AISTATS.*

[22] Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. 2016. Character-aware neural language models. In *Proc. of AAAI.*

[23] Alex Krizhevsky, Geoffrey Hinton, et al. 2009. Learning Multiple Layers of Features from Tiny Images. (2009).

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet Classification with Deep Convolutional Neural Networks. *Proc. of NIPS* (2012).

[25] Fan Lai, Xiangfeng Zhu, Harsha V Madhyastha, and Mosharaf Chowdhury. 2021. Oort: Efficient Federated Learning via Guided Participant Selection. In *Proc. of USENIX OSDI.*

[26] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. 2020. Federated Optimization in Heterogeneous Networks. In *Proc. of MLSys.*

[27] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. 2020. On the Convergence of FedAvg on Non-IID Data. In *Proc. of ICLR.*

[28] Xianfeng Liang, Shuheng Shen, Jingchang Liu, Zhen Pan, Enhong Chen, and Yifei Cheng. 2019. Variance Reduced Local SGD with Lower Communication Complexity. *arXiv preprint arXiv:1912.12844* (2019).

[29] Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. 2020. Ensemble distillation for robust model fusion in federated learning. *Proc. of NeurIPS* (2020).

[30] Amiri M. M., Gunduz D., and Kulkarni S. R. 2020. Federated learning with quantized global model updates. *arXiv preprint arXiv:2006.10672* (2020).

[31] Grigory Malinovskiy, Dmitry Kovalev, Elnur Gasanov, Laurent Condat, and Peter Richtarik. 2020. From local SGD to local fixed-point methods for federated learning. In *Proc. of ICML.*

[32] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proc. of AISTATS.*

[33] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch. In *NIPS-W.*

[34] Reese Pathak and Martin J Wainwright. 2020. FedSplit: An algorithmic framework for fast federated optimization. *Proc. of NeurIPS* (2020).

[35] Hönig R., Zhao Y., and Mullins R. 2022. DAdaQuant: Doubly-adaptive quantization for communication-efficient Federated Learning. In *Proc. of ICML.*

[36] Sashank J. Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečný, Sanjiv Kumar, and Hugh Brendan McMahan. 2021. Adaptive Federated Optimization. In *Proc. of ICLR.*

[37] Monica Ribero and Haris Vikalo. 2020. Communication-Efficient Federated Learning via Optimal Client Sampling. *arXiv preprint arXiv:2007.15197* (2020).

[38] Daniel Rothchild, Ashwinee Panda, Enayat Ullah, Nikita Ivkin, Ion Stoica, Vladimir Braverman, Joseph Gonzalez, and Raman Arora. 2020. FetchSGD: Communication-Efficient Federated Learning with Sketching. In *Proc. of ICML.*

[39] Yichen Ruan, Xiaoxi Zhang, Shu-Che Liang, and Carlee Joe-Wong. 2021. Towards Flexible Device Participation in Federated Learning. In *Proc. of AISTATS.*

[40] Karen Simonyan and Andrew Zisserman. 2015. Very Deep Convolutional Networks for Large-scale Image Recognition. In *Proc. of ICLR.*

[41] Sebastian U Stich and Sai Praneeth Karimireddy. 2020. The error-feedback framework: Better rates for sgd with delayed gradients and compressed updates. *Journal of Machine Learning Research* 21 (2020), 1–36.

[42] Minxue Tang, Xuefei Ning, Yitu Wang, Jingwei Sun, Yu Wang, Hai Li, and Yiran Chen. 2022. FedCor: Correlation-Based Active Client Selection Strategy for Heterogeneous Federated Learning. In *Proc. of IEEE/CVF CVPR.*

[43] Hao Wang, Zakhary Kaplan, Di Niu, and Baochun Li. 2020. Optimizing Federated Learning on Non-IID Data With Reinforcement Learning. In *Proc. of IEEE INFOCOM.*

[44] Hongyi Wang, Mikhail Yurochkin, Yuekai Sun, Dimitris Papailiopoulos, and Yasaman Khazaeni. 2020. Federated Learning with Matched Averaging. In *Proc. of ICLR.*

[45] Jianyu Wang and Gauri Joshi. 2019. Adaptive Communication Strategies to Achieve the Best Error-Runtime Trade-off in Local-update SGD. In *Proc. of SysML.*

[46] Jianyu Wang and Gauri Joshi. 2021. Cooperative SGD: A Unified Framework for the Design and Analysis of Local-Update SGD Algorithms. *Journal of Machine Learning Research* 22, 213 (2021), 1–50.

[47] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. 2020. Tackling the Objective Inconsistency Problem in Heterogeneous Federated Optimization. *Proc. of NeurIPS* (2020).

[48] Jianqiao Wangni, Jialei Wang, Ji Liu, and Tong Zhang. 2018. Gradient sparsification for communication-efficient distributed optimization. *Advances in Neural Information Processing Systems* 31 (2018).

[49] Blake Woodworth, Kumar Kshitij Patel, Sebastian Stich, Zhen Dai, Brian Bullins, Brendan Mcmahan, Ohad Shamir, and Nathan Srebro. 2020. Is local SGD better than minibatch SGD?. In *Proc. of ICML.*

[50] Han Xiao, Kashif Rasul, and Roland Vollgraf. 2017. Fashion-MNIST: A Novel Image Dataset for Benchmarking Machine Learning Algorithms. *arXiv preprint arXiv:1708.07747* (2017).

[51] Guojun Xiong, Gang Yan, Rahul Singh, and Jian Li. 2021. Straggler-resilient distributed machine learning with dynamic backup workers. *arXiv preprint arXiv:2102.06280* (2021).

[52] Gang Yan, Hao Wang, and Jian Li. 2022. Seizing Critical Learning Periods in Federated Learning. In *Proc. of AAAI.*

[53] Gang Yan, Hao Wang, Xu Yuan, and Jian Li. 2023. CriticalFL: A Critical Learning Periods Augmented Client Selection Framework for Efficient Federated Learning. (2023). https://www.dropbox.com/s/m501qs0pppmgu9y/main.pdf?dl=0

[54] Gang Yan, Hao Wang, Xu Yuan, and Jian Li. 2023. DeFL: Defending Against Model Poisoning Attacks in Federated Learning via Critical Learning Periods Awareness. In *Proc. of AAAI.*

[55] Zezhang Yang, Jian Li, and Ping Yang. 2021. Fedadmp: A joint anomaly detection and mobility prediction framework via federated learning. *ICST Transactions on Security and Safety* 8, 29 (2021).

[56] Hao Yu, Sen Yang, and Shenghuo Zhu. 2019. Parallel Restarted SGD with Faster Convergence and Less Communication: Demystifying Why Model Averaging Works for Deep Learning. In *Proc. of AAAI.*