TOWARDS SCALABLE AND NON-IID ROBUST HIERARCHICAL FEDERATED LEARNING VIA LABEL-DRIVEN KNOWLEDGE AGGREGATOR

Anonymous authors
Paper under double-blind review

ABSTRACT

In real-world applications, Federated Learning (FL) meets two challenges: (1) scalability, especially when applied to massive IoT networks, and (2) how to be robust against an environment with heterogeneous data. Realizing the first problem, we aim to design a novel FL framework named Full-stack FL (F2L). More specifically, F2L utilizes a hierarchical network architecture, making extending the FL network accessible without reconstructing the whole network system. Moreover, leveraging the advantages of hierarchical network design, we propose a new label-driven knowledge distillation (LKD) technique at the global server to address the second problem. As opposed to current knowledge distillation techniques, LKD is capable of training a student model, which consists of good knowledge from all teachers' models. Therefore, our proposed algorithm can effectively extract the knowledge of the regions' data distribution (i.e., the regional aggregated models) to reduce the divergence between clients' models when operating under the FL system with non-independent identically distributed data. Extensive experiment results reveal that: (i) our F2L method can significantly improve the overall FL efficiency in all global distillations, and (ii) F2L rapidly achieves convergence as global distillation stages occur instead of increasing on each communication cycle.

1 INTRODUCTION

Recently, Federated Learning (FL) is known as a novel distributed learning methodology for enhancing communication efficiency and ensuring privacy in traditional centralized one McMahan et al. (2017). However, the most challenge of this method for client models is non-independent and identically distributed (non-IID) data, which leads to divergence into unknown directions. Inspired by this, various works on handling non-IID were proposed in Li et al. (2020); Acar et al. (2021); Dinh et al. (2021a); Karimireddy et al. (2020); Wang et al. (2020); Zhu et al. (2021); Nguyen et al. (2022b). However, these works mainly rely on arbitrary configurations without thoroughly understanding the models' behaviors, yielding low-efficiency results. Aiming to fulfil this gap, in this work, we propose a new hierarchical FL framework using information theory by taking a deeper observation of the model's behaviors, and this framework can be realized for various FL systems with heterogeneous data. In addition, our proposed framework can trigger the FL system to be more scalable, controllable, and accessible through hierarchical architecture. Historically, anytime a new segment (i.e., a new group of clients) is integrated into the FL network, the entire network must be retrained from the beginning. Nevertheless, with the assistance of LKD, the knowledge is progressively transferred during the training process without information loss owing to the empirical gradients towards the newly participated clients' dataset.

The main contributions of the paper are summarized as follows. **(1)** We show that conventional FLs performance is unstable in heterogeneous environments due to non-IID and unbalanced data by carefully analyzing the basics of Stochastic Gradient Descent (SGD). **(2)** We propose a new multi-teacher distillation model, Label-Driven Knowledge Distillation (LKD), where teachers can only share the most certain of their knowledge. In this way, the student model can absorb the most meaningful information from each teacher. **(3)** To trigger the scalability and robustness against non-IID data in FL, we propose a new hierarchical FL framework, subbed Full-stack Federated Learning (F2L). Moreover, to guarantee the computation cost at the global server, F2L architecture integrates both techniques: LKD and FedAvg aggregators at the global server. To this end, our framework can do robust training by LKD when the FL process is divergent (i.e., at the start of the training process). When the training starts to achieve stable convergence, FedAvg is utilized to reduce the server's computational cost while retaining the FL performance. **(4)** We theoretically investigate our LKD technique to make a brief comparison in terms of performance with the conventional Multi-teacher knowledge distillation (MTKD), and in-theory show that our new technique always achieves better performance than MTKD. **(5)** We validate the practicability of the proposed LKD and F2L via various experiments based on different datasets and network settings. To show the efficiency of F2L in dealing with non-IID and unbalanced data, we provide a performance comparison and the results show that the proposed F2L architecture outperforms the existing FL methods. Especially, our approach achieves comparable accuracy when compared with FedAvg (McMahan et al. (2017)) and higher $7-20\%$ in non-IID settings.

2 RELATED WORK

2.1 FEDERATED LEARNING ON NON-IID DATA

To narrow the effects of divergence weights, some recent studies focused on gradient regularization aspects Li et al. (2020); Acar et al. (2021); Dinh et al. (2021a); Karimireddy et al. (2020); Wang et al. (2020); Zhu et al. (2021); Nguyen et al. (2022b). By using the same conceptual regularization, the authors in Li et al. (2020); Acar et al. (2021), and Dinh et al. (2021a) introduced the FedProx, FedDyne, and FedU, respectively, where FedProx and FedDyne focused on pulling clients' models back to the nearest aggregation model while FedU's attempted to pull distributed clients together. To direct the updated routing of the client model close to the ideal server route, the authors in Karimireddy et al. (2020) proposed SCAFFOLD by adding a control variate to the model updates. Meanwhile, to prevent the aggregated model from following highly biased models, the authors in Wang et al. (2020) rolled out FedNova by adding gradient scaling terms to the model update function. Similar to Dinh et al. (2021a), the authors in Nguyen et al. (2022b) launched the WALF by applying Wasserstein metric to reduce the distances between local and global data distributions. However, all these methods are limited in providing technical characteristics. For example, Wang et al. (2020) demonstrated that FedProx and FedDyne are ineffective in many cases when using pullback to the globally aggregated model. Meanwhile, FedU and WAFL have the same limitation on making a huge communication burden. Aside from that, FedU also faces a very complex and non-convex optimization problem.

Regarding the aspect of knowledge distillation for FL, only the work in Zhu et al. (2021) proposed a new generative model of local users as an alternative data augmentation technique for FL. However, the majority drawback of this model is that the training process at the server demands a huge data collection from all users, leading to ineffective communication.

Motivated by this, we propose a new FL architecture that is expected to be more elegant, easier to implement, and much more straightforward. Unlike Dinh et al. (2021a); Acar et al. (2021); Karimireddy et al. (2020), we utilize the knowledge from clients' models to extract good knowledge for the aggregation model instead of using model parameters to reduce the physical distance between distributed models. Following that, our proposed framework can flexibly handle weight distance and probability distance in an efficient way, i.e., $\|p^k(y=c)-p(y=c)\|$ (please refer to Appendix B).

2.2 MULTI-TEACHER KNOWLEDGE DISTILLATION

MTKD is an improved version of KD (which is presented in Appendix A.2), in which multiple teachers work cooperatively to build a student model. As shown in Zhu et al. (2018), every MTKD technique solves the following problem formulation:

$$\mathbf{P1}: \min \mathcal{L}_m^{KL} = \sum_{r=1}^{R}\sum_{l=1}^{C} \hat{p}(l|\boldsymbol{X},\boldsymbol{\omega}^r,T)\log\frac{\hat{p}(l|\boldsymbol{X},\boldsymbol{\omega^r},T)}{\hat{p}(l|\boldsymbol{X},\boldsymbol{\omega}^g,T)}, \tag{1}$$

here, $r\in\{R\}$ are the teachers' indices. By minimizing **P1**, the student $\hat{p}^g$ can attain knowledge from all teachers. However, when using MTKD, there are some problems in extracting the knowledge distillation from multiple teachers. In particular, the process of distilling knowledge in MTKD is typically empirical without understanding the teacher's knowledge (i.e., aggregating all KL divergences

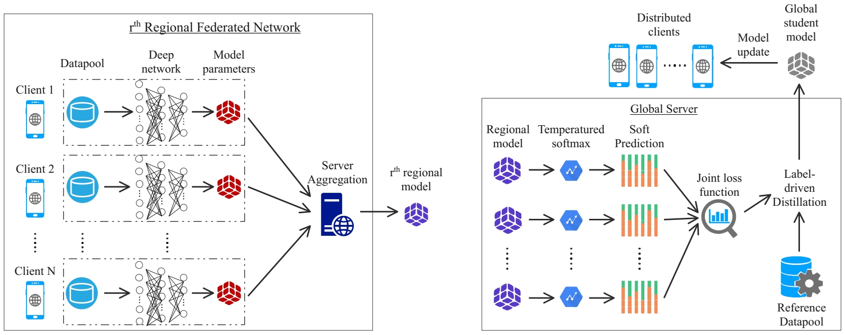


Figure 1: The architecture of our F2L framework.

between each teacher and the student). Therefore, MTKD is unable to exploit teachers' detailed predictions for the KD (e.g., Liu et al. (2020c), Asif et al. (2019), Zhu et al. (2018), Fukuda et al. (2017), Tran et al. (2020)). Another version of MTKD, KTMDs can only apply for a better teachers to distill knowledge (e.g., Shen et al. (2019), Zhu & Wang (2021), Zhang et al. (2022), Son et al. (2021)). For example, as provided in (Shen et al., 2019, eq. 6), the student only selects the best teacher to operate the knowledge distillation. Visually, this technique is the same as the way of selecting a teacher among a set of teachers to carry out a single teacher distillation. Therefore, the student's performance is always bounded by the best teacher's performance. Another popular research direction in MTKD is to leverage the advantage of the gap between teachers' hidden class features. However, owing to the lack of explanatory knowledge in teachers' hidden layers, the method in Zhu & Wang (2021) cannot obtain better student performance when compared to their teachers. Generally, current MTKD techniques cannot extract good knowledge from different customer models, leading to weight divergence in FL.

3 FULL-STACK FEDERATED LEARNING

3.1 THE F2L FRAMEWORK

The main objective of our work is to design a hierarchical FL framework, in which a global server manages a set of distinct regional servers. Utilizing hierarchical FL, our proposed algorithm can achieve computation and computation efficiency. The reason is that Hierarchical FL makes the clients to train sectionally before making the global aggregation Liu et al. (2020a); Briggs et al. (2020). Consequently, FL inherits every advantage from mobile edge intelligence concept over traditional non-hierarchical networks (e.g., communication efficiency, scalability, controlability) Pham et al. (2020); Luong et al. (2016); Lim et al. (2020). At the end of each knowledge-sharing episode, the regions (which are supervised by regional servers) cooperate and share their knowledge (each region functions as a distinguished FL system, with a set amount of communication rounds per episode).

In each episode, each region randomly selects a group of clients from the network to carry out the aggregation process (e.g., FedAvg, FedProx); therefore, each region functions as a small-scale FL network. As a result, there are always biases in label-driven performance by applying random sampling on users per episode (see Appendix F). Given the random sampling technique applied to the regional client set, the regions always have different regional data distributions. Consequently, various label-driven performances of different teachers might be achieved.

At the global server, our goal is to extract good performance from regional teachers while maintaining the salient features (e.g., understanding of the regional data distributions) of all regions. As a result, we can capture useful knowledge from diverse regions in each episode using our proposed innovative knowledge distillation technique (which is briefly demonstrated in Section 3.2). We train the model on the standard dataset on the central server to extract knowledge from multiple teachers into the global student model. The preset data pool on the server $\mathcal{S}$ is used to verify the model-class reliability and generate pseudo labels.

The system model is illustrated in Fig. 1, and thoroughly described in Appendix C. The pseudo algorithm for F2L is demonstrated in Algorithm 1. When the FL process suffers from client-drift Karimireddy et al. (2020) (i.e., the distribution of label-driven accuracies of different regions have large variance), the F2L framework applies LKD to reduce the class-wise performance gaps between regions (i.e., the regions with better performance on a particular class share their knowledge to regions with low performance). As a result, the FL network achieves a significantly faster convergence when utilizing LKD (which is intensively explained in Section 3.2.) for the global aggregation process. When the generalization gap between regions is considerably reduced (i.e., $\| \max_r \beta_r^c - \min_r \beta_r^c \| \leq \epsilon$), our F2L network becomes vanilla FL to reduce computation and communication costs. To this end, our method can achieve computation efficiency while showing robustness to the non-IID data in the network. Additionally, whenever a new set of clients are added into the network and makes positive contributions to the FL system (e.g., $\| \max_r \beta_r^c - \min_r \beta_r^c \| \geq \epsilon$ where $\| \max_r \beta_r^c \|$ a corresponding to the new region's performance) the LKD aggregator can be switched back to improve the FL system's performance over again.

3.2 LABEL-DRIVEN KNOWLEDGE DISTILLATION

To extract knowledge from multiple teachers to the global student model, we train the model on the standard dataset on the central server. The preset data pool on the server $\mathcal{S}$ is used to verify the model-class reliability and generate pseudo labels. In our work, the MTKD process executes two main tasks: (1) extracting the teachers' knowledge and (2) maintaining the students' previous performance.

To comprehend the LKD method, we first revisit the conventional MTKD, where the probabilistic output is calculated by model $\boldsymbol{\omega}$ on x_i, the prediction label c is $\hat{p}(l|x_i, \boldsymbol{\omega}, T, c)$ and its relation is:

$$\hat{p}(l|x_i, \boldsymbol{\omega}, T, c) = \begin{cases} \hat{p}(l|x_i, \boldsymbol{\omega}, T), & \text{if argmax } [\hat{p}(l|x_i, \boldsymbol{\omega}, T)] = c, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

On the one hand, we aim to transfer knowledge from different regional models to the global one. Inspired by Hinton et al. (2015), we use the Kullback–Leibler (KL) divergence between each regional teacher and the global logits as a method to estimate the difference between two models' performance. The relationship is expressed as follows:

$$\mathcal{L}_r^{KL} = \sum_{c=1}^{C} \beta_r^c \sum_{i=1}^{S_c^r} \sum_{l=1}^{C} \hat{p}^r(l|x_i, \boldsymbol{\omega}^r, T, c) \times \log \frac{\hat{p}^r(l|x_i, \boldsymbol{\omega}^r, T, c)}{\hat{p}^g(l|x_i, \boldsymbol{\omega}^g, T, c)}, \tag{3}$$

where S is the number of samples of the fixed dataset $\mathcal{S}$ on the server. $(\boldsymbol{X}_{\text{alg}}^r, \boldsymbol{Y}_{\text{alg}}^r)$ is the dataset which is pseudo labeled and aligned by regional model r and $(\boldsymbol{X}_{\text{alg}}^r[c], \boldsymbol{Y}_{\text{alg}}^r[c])$ represents the set of data with size of S_c^r labeled by the model r as c. Although the same preset dataset is utilized on every teacher model, the different pseudo labeling judgments from different teachers lead to the different dataset tuples. The process of identifying S_c^r is demonstrated in Algorithm 3. Because the regional models label on the same dataset S, we have $\sum_{c=1}^{C} S_c^r = S$ for all regional models. $D_{KL}^c(\hat{p}^r \| \hat{p}^g)$ is the c label-driven KL divergence between model r and model g.

On the other hand, we aim to guarantee that the updated global model does not forget the crucial characteristics of the old global model. Hence, to measure the divergence between the old and the updated model, we introduce the following equation:

$$\mathcal{L}_{\boldsymbol{\omega}_{upd}}^{KL} = \sum_{c=1}^{C} \beta_{\boldsymbol{\omega}_{old}}^c \sum_{i=1}^{S_c^r} \sum_{l=1}^{C} \hat{p}^g(l|x_i, \boldsymbol{\omega}_{old}^g, T, c) \times \log \frac{\hat{p}^g(l|x_i, \boldsymbol{\omega}_{old}^g, T, c)}{\hat{p}^g(l|x_i, \boldsymbol{\omega}_{new}^g, T, c)}, \tag{4}$$

where $\boldsymbol{\omega}_{old}$ is the old parameters set of the global model which is distilled in the last episode of F2L. More details about the label-driven knowledge distillation are discussed in Appendix G.

To compare the performance between LKD and MTKD, we consider the following assumption and lemmas:

Lemma 1 *Given τ_r^c is the c-label driven predicting accuracy on model r. Let $\sigma_{r,c}^2, \mu_{r,c}$ be the model's variance and mean, respectively. The optimal value of variance and mean on student model (i) $\sigma_{LKD,g,c}^{*2}, \mu_{LKD,g,c}^{*}$ yields $\sigma_{LKD,g,c}^{*2} = \frac{1}{\sum_{r=1}^{R} e^{\tau_r^c}} \sum_{r=1}^{R} e^{\tau_r^c} \sigma_{r,c}^2$, and $\mu_{LKD,g,c}^{*} = \frac{1}{\sum_{r=1}^{R} e^{\tau_r^c}} \sum_{r=1}^{R} e^{\tau_r^c} \mu_{r,c}$..*

Proof: The proof is provided in Appendix J.

Assumption 1 *Without loss of generality, we consider R distinct regional models whose accuracy satisfy the following prerequisites $\sigma^2_{1,c} \leq \sigma^2_{2,c} \leq \ldots \leq \sigma^2_{R,c}$, and $|\mu_{1,c} - \bar{\mu}_c| \leq |\mu_{2,c} - \bar{\mu}_c| \leq \ldots \leq |\mu_{R,c} - \bar{\mu}_c|$ ($\bar{\mu}_c$ is denoted as an empirical global mean of the dataset on class c).*

Lemma 2 *Given the set of models with variance satisfy $\sigma^2_{1,c} \leq \sigma^2_{2,c} \leq \ldots \leq \sigma^2_{R,c}$, the models' accuracy have the following relationship $\tau^c_1 \geq \tau^c_2 \geq \ldots \geq \tau^c_R$.*

Proof. The proof can be found in Appendix K.

Theorem 1 *Let $\sigma^{*2}_{LKD,g,c}$ be the class-wise variance of the student model, and $\sigma^{*2}_{MTKD,g,c}$ be the class-wise variance of the model of teacher r, respectively. We always have the student's variance using LKD technique always lower than that using MTKD:*

$$\sigma^{*2}_{LKD,g,c} \leq \sigma^{*2}_{MTKD,g,c}. \tag{5}$$

Proof: For the complete proof see Appendix H.

Theorem 2 *Let $\mu^*_{LKD,g,c}$ be the empirical c-class-wise mean of the student model, and $\mu^*_{MTKD,g,c}$ be the empirical c-class-wise mean of the model of teacher r, respectively. We always have the student's empirical mean using LKD technique always closer to the empirical global dataset's class-wise mean ($\bar{\mu}_c$) than that using MTKD:*

$$|\mu^*_{LKD,g,c} - \bar{\mu}_c| \leq |\mu^*_{MTKD,g,c} - \bar{\mu}_c|. \tag{6}$$

Given Theorems 1 and 2, we can prove that our proposed LKD technique can consistently achieve better performance than that of the conventional MTKD technique. Moreover, by choosing the appropriate LKD allocation weights, we can further improve the LKD performance over MTKD. Due to space limitation, we defer the proof to Appendix I.

3.3 CLASS RELIABILITY SCORING

The main idea of class reliability variables β^c_r, $\beta^c_{\boldsymbol{\omega}_{old}}$ in LKD is to weigh the critical intensity of the specific model. Therefore, we leverage the attention design from Vaswani et al. (2017) to improve the performance analysis of teachers' label-driven.

For regional models with disequilibrium or non-IID data, the teachers only teach the predictions relying upon their specialization. The prediction's reliability can be estimated by leveraging the validation dataset on the server and using the function under the curve (AUC) as follows:

$$\beta^c_r = \frac{\exp(f^{c,r}_{AUC} T_{\boldsymbol{\omega}})}{\sum^R_{r=1} \exp(f^{c,r}_{AUC} T_{\boldsymbol{\omega}})}, \tag{7}$$

where $f^{c,r}_{\text{AUC}}$ denotes the AUC function on classifier c of the regional model r. Since AUC provides the certainty that a specific classifier can work on a label over the rest, we use the surrogate softmax function to weigh the co-reliability among the same labeling classifiers on different teacher models. For simplicity, we denote $\beta^c_{\boldsymbol{\omega}_{old}}$ as the AUC on each labeling classifier:

$$\beta^c_{\boldsymbol{\omega}_{old}} = \frac{\exp(f^{c,\boldsymbol{\omega}_{\text{old}}}_{AUC} T_{\boldsymbol{\omega}})}{\exp(f^{c,\boldsymbol{\omega}_{\text{new}}}_{AUC} T_{\boldsymbol{\omega}}) + \exp(f^{c,\boldsymbol{\omega}_{\text{old}}}_{AUC} T_{\boldsymbol{\omega}})}. \tag{8}$$

In the model update class reliability, instead of calculating the co-reliability between teachers, equation 8 compares the performance of the previous and current global models. Moreover, we introduce a temperatured value for the class reliability scoring function, denoted as $T_{\boldsymbol{\omega}}$. By applying a large temperatured value, the class reliability variable sets β^c_r, and $\beta^c_{\boldsymbol{\omega}_{old}}$ make a higher priority on the better performance (i.e., the label-driven performance on class c from teacher r, e.g., $f^{c,r}_{AUC}$ in equation equation 7 or class c from old model $\boldsymbol{\omega}_{\text{old}}$ in equation equation 8). By this way, we can preserve the useful knowledge which is likely ignored in the new distillation episode. The more detailed descriptions of class reliability scoring are demonstrated in Algorithm 6.

3.4 JOINT MULTI-TEACHER DISTILLATION FOR F2L

We obtain the overall loss function for online distillation training by the proposed F2L:

$$\mathcal{L}_{\text{F2L}} = \lambda_1 \sum_{r=1}^{R} \mathcal{L}_r^{KL} + \lambda_2 \mathcal{L}_{\boldsymbol{\omega}_{upd}}^{KL} + \lambda_3 \mathcal{L}_{CE}^{g}, \tag{9}$$

where $\lambda_1, \lambda_2, \lambda_3$ are the scaling coefficients of the three terms in the joint loss function. The first and second terms imply the joint LKD from the regional teacher models and the updating correction step, respectively. Moreover, to ensure that knowledge the student receives from teachers is accurate and can be predicted accurately in practice, we need to validate the quality of the student model on the real data. Thus, we also compute the "standard" loss between the student and the ground-truth labels of the train dataset. This part is also known as the hard estimator, which is different from the aforementioned soft-distillator. The hard loss equation is as follows:

$$\mathcal{L}_{CE}^{g} = H(y, \hat{p}(l|\boldsymbol{X}, \boldsymbol{\omega}^g, T)) = \sum_{l=1}^{C} y_l \log \hat{p}(l|\boldsymbol{X}, \boldsymbol{\omega}^g, T). \tag{10}$$

Algorithm 1 F2L framework

```
Require: Initialize clients' weights, global aggregation round, number of regions $R$, arbitrary $\epsilon$.
while not converge do
    for all regions $r \in \{1, 2, \ldots, R\}$ do
        for all user in regions do
            Apply FedAvg on regions $r$.
        end for
        Send regional model $\boldsymbol{\omega}^r$ to the global server.
    end for
    if reach global aggregation round then
        if $\| \max_r \beta_r^c - \min_r \beta_r^c \| \geq \epsilon$ where $\beta = \{\beta_r^1, \ldots, \beta_r^C\}|_{r=1}^{R}$ from Algorithm 6 then
            Apply LKD as described in Algorithm 2
        else
            $\boldsymbol{\omega}^g = 1/R \sum_{r=1}^{R} \boldsymbol{\omega}^r$.
        end if
    end if
end while
```

We use the temperature coefficient $T = 1$ to calculate the class probability for this hard loss. The overall training algorithm for LKD is illustrated in Algorithm 2. In terms of value selection for scaling coefficients, the old global model can be considered as an additional regional teacher's model in the same manner, in theory. Therefore, λ_2 should be chosen as:

$$\lambda_2 = \begin{cases} \frac{1}{R}\lambda_1, & \text{if update distillation in equation 4 is considered,} \\ 0, & \text{otherwise,} \end{cases} \tag{11}$$

where R is the number of regions decided by our hierarchical FL settings. With respect to λ_3, the value is always set as:

$$\lambda_3 = \begin{cases} 1 - \frac{R+1}{R}\lambda_1, & \text{if update distillation in equation 4 is considered,} \\ 1 - \lambda_1, & \text{otherwise.} \end{cases} \tag{12}$$

3.5 DISCUSSIONS ON THE EXTENT OF PROTECTING PRIVACY

In its simplest version, our proposed F2L framework, like the majority of existing FL approaches, necessitates the exchange of models between the server and each client, which may result in privacy leakage due to, for example, memorization present in the models. Several existing protection methods can be added to our system in order to safeguard clients against enemies. These include adding differential privacy Geyer et al. (2017) to client models and executing hierarchical and decentralized model fusion by synchronizing locally inferred logits, for example on random public data, as in work Chang et al. (2019). We reserve further research on this topic for the future.

Table 1: The top-1 test accuracy of different baselines on different data settings. The α indicates the non-IID degree of the dataset (the lower value of α means that the data is more heterogeneous).

Dataset	FedAvg	FedGen	FedProx	Fed-Distill	F2L (Ours)	FedAvg	FedGen	FedProx	Fed-Distill	F2L (Ours)
	Dirichlet ($\alpha = 1$)					Dirichlet ($\alpha = 0.1$)				
EMNIST	71.66	78.70	70.77	75.56	**81.14**	59.10	68.24	58.88	46.03	**68.31**
CIFAR-10	60.48	59.21	63.72	62.36	**71.22**	47.07	47.08	47.05	45.67	**55.22**
CIFAR-100	36.17	40.26	36.3	34.88	**50.33**	21.31	28.96	20.43	16.15	**31.07**
CINIC-10	65.23	71.61	65.15	67.77	**74.85**	47.55	52.35	48.2	47.1	**57.12**
CelebA	70.82	75.43	71.07	68.59	**81.65**	63.58	70.14	66.33	62.91	**74.14**

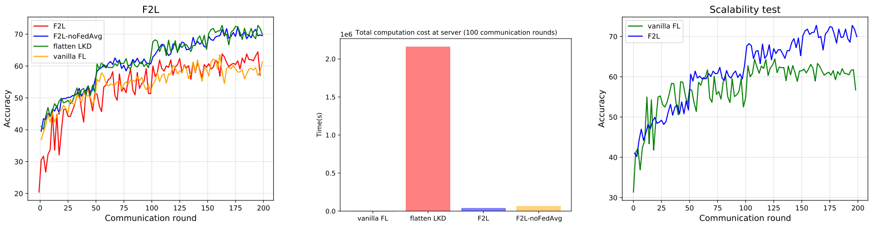
4 EXPERIMENTAL EVALUATION

4.1 COMPARISON WITH FL METHODS

We run the baselines (see Section E) and compare with our F2L. Then, we evaluate the comparisons under different non-IID ratio. More precisely, we generate the IID data and non-IID data with two different Dirichlet balance ratio: $\alpha = \{1, 10\}$. The comparison results are presented in Table 1. As shown in Table 1, the F2L can outperform the four baselines with a significant increase in accuracy. The reason for this phenomenon is that the LKD technique selectively draws the good features from regional models to build a global model. Hence, the global model predicts the better result on each different class and the entire accuracy of the global model then increases tremendously. The significant impact when applying LKD to distill different teachers to one student is shown in Table 2.

4.2 COMPUTATION EFFICIENCY OF F2L

To evaluate the computation efficiency of our proposed F2L process, we compare our F2L process with 3 benchmarks: (i) F2L-noFedAvg (aggregator only consists of LKD), (ii) vanilla FL (FL with flatten architecture and FedAvg as an aggregator), and (iii) flatten LKD (FL with flatten architecture based with LKD as an alternate aggregator). Fig. 2(a) shows that the F2L system can achieve convergence as good as the F2L-noFedAvg. The reason is that: after several communication rounds, the distributional distance between regions is reduced thanks to the LKD technique. Hence, the efficiency of the LKD technique on the data is decreased. As a consequence, the LKD technique shows no significant robustness over FedAvg aggregator. In the non-hierarchical settings, the flatten LKD and FedAvg reveal under-performed compared to the proposed hierarchical settings. We assume that the underperformance above comes from the data shortage of clients' training models. To be more detailed, the clients' dataset are considerably smaller than that of the "regional dataset". Thus, the regional models contain more information than the clients' models. We believe that: in the LKD technique, teachers' models require a certain amount of knowledge to properly train a good student (i.e., the global model). Given the convergence rate from Fig. 2(a) and the computation cost at the



(a) (b) (c)

Figure 2: Performance benchmarks of F2L under different settings. Fig. 2(a) reveals the convergence. Fig. 2(b) shows the computational cost, and Fig. 2(c) demonstrates the F2L convergence when a new set of clients are added into the FL system (i.e., at communication round 100).

Table 2: Top-1 accuracy of F2L on 5 datasets MNIST, EMNIST, CIFAR-100, , CINIC-10 and CelebA. The data's heterogeneity is set at $\alpha = 0.1$ on CIFAR-100, MNIST, CINIC-10 and CelebA. We use EMNIST "unbalanced" to evaluate in this test. The "before update" and "after update" denote the teacher models' accuracies before and after the global distillation, respectively.

	MNIST		EMNIST		CIFAR-100		CINIC-10		Celeb-A	
	before	after	before	after	before	after	before	after	before	after
	update	update	update	update	update	update	update	update	update	update
Teacher 1	61.02	**95.19**	73.27	**84.09**	20.11	**35.41**	43.8	**46.59**	62.37	**67.98**
Teacher 2	92.49	**98.22**	78.80	**83.62**	18.82	**31.2**	42.15	**46.01**	63.79	**72.33**
Teacher 3	81.60	**97.63**	80.5	**84.10**	22.40	**34.93**	40.02	**42.15**	64.05	**69.44**
G-student	**98.71**		**84.11**		**37.68**		**47.65**		**70.12**	

server on Fig. 2(b), we can see that, by using the adaptive switch between LKD and FedAvg in F2L, we can achieve significant computational efficiency at the aggregation server. Note that F2L can substantially increase performance and computation efficiency compared with non-hierarchical architecture.

4.3 SCALABILITY

This section evaluates the F2L scalability. To do so, we inject a group of clients with non-IID data into our FL system after 100 rounds (when the convergence becomes stable). Note that the FL system has never trained these data. The detailed configurations of our experimental assessments can be found in Appendix E. As it can be seen from Fig. 2(c), when a new group of clients are added to the FL system, the vanilla FL shows a significant drop in terms of convergence. The reason is because of the distribution gap between the global model's knowledge and knowledge of the clients' data. Whenever new data with unlearned distribution is added to a stable model, the model will make considerable gradient steps towards the new data distribution. Thus, the FedAvg takes considerable learning steps to become stable again. In contrast, in F2L system, the learning from newly injected regions does not directly affect the learning of the whole FL system. Instead, the knowledge from the new domains is selectively chosen via the LKD approach. Thus, the LKD process does not suffer from information loss when new clients with non-IID data are added to the FL system.

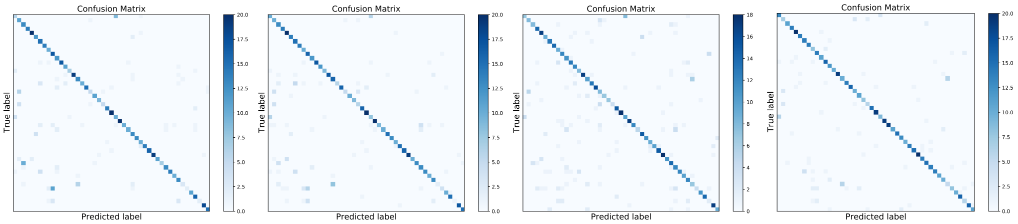
4.4 LKD ANALYSIS

In this section, we evaluate the LKD under various settings to justify the capability of LKD to educate the good student from the normal teachers. Our evaluations are as follows.

Can student outperform teachers? To verify the efficiency of LKD with respect to enhancing student performance, we first evaluate F2L on MNIST, EMNIST, CIFAR-100, CINIC-10, CelebA dataset. The regions are randomly sampled from the massive FL network. In this way, we only evaluate the performance of LKD on random teachers. Table 2 shows top-1 accuracy on the regional teacher and student models. The results reveal that LKD can significantly increase the global model performance compared with that of the regional models. Moreover, the newly distilled model can work well under each regional non-IID data after applying the model update.

To make a better visualization for the LKD's performance, we reveal the result of LKD on EMNIST dataset in terms of confusion matrix as in Fig. 3. As it can be seen from the figure, the true predictions is represented by a diagonals of the matrices. A LKD performance is assumed to be well predicted when the value on diagonals is high (i.e., the diagonals' colors is darker), and the off-diagonals is low (i.e., the off-diagonals' colors are lighter). As we can see from the four figures, there are a significant reduce in the off-diagonals' darkness in the student performance (i.e., Fig. 3(d)) comparing to the results in other teachers (i.e., Figures 3(a), 3(b), and 3(c)). Therefore, we can conclude that our proposed MTKD techniques can surpass the teachers' performance as we mentioned in Section 2.

Teachers can really educate student? We evaluate LKD under different soft-loss coefficients λ_1 while the hard-loss factor is set at $\lambda_3 = 1 - \lambda_1$ (the scaling value λ_2 is set to 0). Thus, we can justify whether the robust performance of LKD comes from the joint distillation from teachers or just the exploitation of data-on-server training. We evaluate LKD on six scaling values



(a) (b) (c) (d)

Figure 3: The illustrative results of LKD on EMNIST dataset. Confusion matrices show the effectiveness of joint distillation on regional models. Figures (a), (b), and (c) are the confusion matrix before distillation of teacher's predictions in region 1, 2, and 3, respectively (see Appendix E). Fig. (d) is the confusion matrix of predictions after distillation of student. The matrix diagonal demonstrates the true-predicted label of the model.

$\lambda_1 = \{0, 0.001, 0.01, 0.1, 0.5, 1\}$. We evaluate on three dataset, including EMNIST, CIFAR-10, and CIFAR-100, and summarize the results in Tables 5, 6 and 7 in **Appendices**. We can see from the three tables that the LKD cap off with $\lambda_3 = 0.01$. Especially, when $\lambda_3 = 1$ (which means the LKD acts as a vanilla cross-entropy optimizer), the model accuracy reduces notably. This means that the LKD only uses hard-loss as a backing force to aid the distillation. Thus, our LKD is appropriate and technically implemented.

Required training sample size for joint distillation. To justify the ability of LKD under a shortage of training data, we evaluate LKD with six different data-on-server settings: $\sigma = \{1, 1/2, 1/4, 1/6, 1/8, 1/10\}$, where σ is the sample ratio when compared with the original data-on-server as demonstrated in Table 4. As we can see from the implementation results in three Tables 8, 9, and 10 in **Appendices**, the F2L is demonstrated to perform well under a relatively small data-on-server. To be more specific, we only need the data-on-server to be 4 times lower than the average data-on-client to achieve a robust performance compared with the vanilla FedAvg. However, we suggest using the data-on-server to be larger than the data from distributed clients to earn the highest performance for LKD. Moreover, due to the ability to work under unlabeled data, the data-on-server does not need to be all labeled. We only need a small amount of labeled data to aid the hard-loss optimizer. Thus, the distillation data on the server can be updated from distributed clients gradually.

BROADER IMPACT AND LIMITATION

Due to the hierarchical framework of our proposed F2L, each sub-region acts like an independent FL process. Therefore, our F2L is integrable with other current methods, which means that we can apply varying FL techniques (e.g., FedProx, FedDyne, FedNova, HCFL Nguyen et al. (2022a)) into distinct sub-regions to enhance the overall F2L framework. Therefore, architecture search (e.g., which FL technique is suitable for distinct sub FL region) for the entire hierarchical network is essential for our proposed framework, which is the potential research for the future work. Moreover, the hierarchical framework remains unearthed. Therefore, a potentially huge amount of research directions is expected to be investigated (e.g., resource allocation Nguyen et al. (2022c); Saputra et al. (2022; 2021); Dinh et al. (2021b), and task offloading in hierarchical FL Yang et al. (2021)). However, our LKD technique still lacks of understanding about the teachers' models (e.g., how classification boundaries on each layer impact on the entire teachers' performance). By investigating in explainable AI, along with layer-wise performance, we can enhance the LKD, along with reducing the unlabeled data requirements for the distillation process in the future work.

5 CONCLUSION

In this research, we have proposed an FL technique that enables knowledge distillation to extract the-good-feature-only from clients to the global model. Our model is capable of tackling the FL's heterogeneity efficiently. Moreover, experimental evaluations have revealed that our F2L model outperforms all of the state-of-the-art FL baselines in recent years.

REFERENCES

Entropy, Relative Entropy and Mutual Information, chapter 2, pp. 12–49. 2005. ISBN 9780471200611.

Durmus Alp Emre Acar, Yue Zhao, Ramon Matas, Matthew Mattina, Paul Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization. In *International Conference on Learning Representations*, Dec. 2021.

Umar Asif, Jianbin Tang, and Stefan Harrer. Ensemble knowledge distillation for learning improved and efficient networks. *arXiv preprint arXiv:1909.08097*, 2019.

Sergey Bobkov, Mokshay Madiman, and Liyao Wang. Fractional generalizations of young and brunn-minkowski inequalities. Jun. 2011.

Christopher Briggs, Zhong Fan, and Peter Andras. Federated learning with hierarchical clustering of local updates to improve training on non-iid data. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–9, 2020.

Hongyan Chang, Virat Shejwalkar, Reza Shokri, and Amir Houmansadr. Cronus: Robust and heterogeneous collaborative learning with black-box knowledge transfer. *arXiv preprint arXiv:1912.11279*, 2019.

Hong-You Chen and Wei-Lun Chao. Fedbe: Making bayesian model ensemble applicable to federated learning. *arXiv preprint arXiv:2009.01974*, 2020.

Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André van Schaik. EMNIST: an extension of MNIST to handwritten letters, Mar. 2017.

Luke N. Darlow, Elliot J. Crowley, Antreas Antoniou, and Amos J. Storkey. Cinic-10 is not imagenet or cifar-10. Oct. 2018. URL `https://arxiv.org/abs/1810.03505`.

W DelPozzo, C P L Berry, A Ghosh, T S F Haines, L P Singer, and A Vecchio. Dirichlet process Gaussian-mixture model: An application to localizing coalescing binary neutron stars with gravitational-wave observations. *Monthly Notices of the Royal Astronomical Society*, 479(1): 601–614, 06 2018. ISSN 0035-8711.

Canh T Dinh, Tung T Vu, Nguyen H Tran, Minh N Dao, and Hongyu Zhang. FedU: A unified framework for federated multi-task learning with laplacian regularization. *arXiv preprint arXiv:2102.07148*, 2021a.

Thinh Quang Dinh, Diep N. Nguyen, Dinh Thai Hoang, Pham Tran Vu, and Eryk Dutkiewicz. Enabling large-scale federated learning over wireless edge networks. In *2021 IEEE Global Communications Conference (GLOBECOM)*, pp. 01–06, 2021b.

Rick Durrett. *Probability: Theory and Examples*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 4 edition, 2010.

T. Fukuda, Masayuki Suzuki, Gakuto Kurata, Samuel Thomas, Jia Cui, and Bhuvana Ramabhadran. Efficient knowledge distillation from an ensemble of teachers. pp. 3697–3701, 08 2017.

Robin C Geyer, Tassilo Klein, and Moin Nabi. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557*, 2017.

Haowei He, Gao Huang, and Yang Yuan. Asymmetric valleys: Beyond sharp and flat local minima. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., Dec. 2019.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv:1503.02531*, Mar. 2015.

Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335*, 2019.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. SCAFFOLD: Stochastic controlled averaging for federated learning. In *Proceedings of the 37th International Conference on Machine Learning*, 13–18 Jul. 2020.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836*, 2016.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, Aug. 2009.

Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551, Dec. 1989.

Yann LeCun and Corinna Cortes. MNIST handwritten digit database, 2010. URL `http://yann.lecun.com/exdb/mnist/`.

Qinbin Li, Yiqun Diao, Quan Chen, and Bingsheng He. Federated learning on non-IID data silos: An experimental study. In *IEEE 38th International Conference on Data Engineering (ICDE)*, pp. 965–978, 2022.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2:429–450, 2020.

Wei Yang Bryan Lim, Nguyen Cong Luong, Dinh Thai Hoang, Yutao Jiao, Ying-Chang Liang, Qiang Yang, Dusit Niyato, and Chunyan Miao. Federated learning in mobile edge networks: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 22(3):2031–2063, 2020.

Lumin Liu, Jun Zhang, S.H. Song, and Khaled B. Letaief. Client-edge-cloud hierarchical federated learning. In *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, pp. 1–6, 2020a.

Shengchao Liu, Dimitris Papailiopoulos, and Dimitris Achlioptas. Bad global minima exist and sgd can reach them. *Advances in Neural Information Processing Systems*, 33:8543–8552, 2020b.

Yuang Liu, Wei Zhang, and Jun Wang. Adaptive multi-teacher multi-level knowledge distillation. *Neurocomputing*, 415:106–113, 2020c.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, Dec. 2015.

Nguyen Cong Luong, Dinh Thai Hoang, Ping Wang, Dusit Niyato, Dong In Kim, and Zhu Han. Data collection and wireless communication in internet of things (IoT) using economic analysis and pricing models: A survey. *IEEE Communications Surveys & Tutorials*, 18(4):2546–2590, 2016.

David J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Copyright Cambridge University Press, 2003.

H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data, 2017.

Konstantin Mishchenko, Eduard Gorbunov, Martin Takáč, and Peter Richtárik. Distributed learning with compressed gradient differences. *arXiv preprint arXiv:1901.09269*, 2019.

Arvind Neelakantan, Luke Vilnis, Quoc V. Le, Ilya Sutskever, Lukasz Kaiser, Karol Kurach, and James Martens. Adding gradient noise improves learning for very deep networks, Nov. 2015.

Minh-Duong Nguyen, Sang-Min Lee, Quoc-Viet Pham, Dinh Thai Hoang, Diep N. Nguyen, and Won-Joo Hwang. HCFL: A high compression approach for communication-efficient federated learning in very large scale IoT networks. *IEEE Transactions on Mobile Computing*, pp. 1–13, Jun. 2022a.

Tung-Anh Nguyen, Tuan Dung Nguyen, Long Tan Le, Canh T Dinh, and Nguyen H Tran. On the generalization of Wasserstein robust federated learning. *arXiv preprint arXiv:2206.01432*, 2022b.

Xuan-Tung Nguyen, Minh-Duong Nguyen, Quoc-Viet Pham, Vinh-Quang Do, and Won-Joo Hwang. Resource allocation for compression-aided federated learning with high distortion rate. *arXiv preprint arXiv:2206.06976*, 2022c.

Quoc-Viet Pham, Fang Fang, Vu Nguyen Ha, Md. Jalil Piran, Mai Le, Long Bao Le, Won-Joo Hwang, and Zhiguo Ding. A survey of multi-access edge computing in 5G and beyond: Fundamentals, technology integration, and state-of-the-art. *IEEE Access*, 8:116974–117017, 2020.

Herbert Robbins and Sutton Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407, Sep. 1951.

Yuris Mulya Saputra, Diep N. Nguyen, Dinh Thai Hoang, and Eryk Dutkiewicz. Incentive mechanism for ai-based mobile applications with coded federated learning. In *2021 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, 2021.

Yuris.Mulya. Saputra, Diep Nguyen, Hoang.Thai. Dinh, Quoc-Viet Pham, Eryk Dutkiewicz, and Won-Joo Hwang. Federated learning framework with straggling mitigation and privacy-awareness for ai-based mobile application services. *IEEE Transactions on Mobile Computing*, pp. 1–1, 2022.

Chengchao Shen, Mengqi Xue, Xinchao Wang, Jie Song, Li Sun, and Mingli Song. Customizing student networks from heterogeneous teachers via adaptive knowledge amalgamation. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information. *arXiv preprint arXiv:1703.00810*, 2017.

Samuel Smith, Erich Elsen, and Soham De. On the generalization benefit of noise in stochastic gradient descent. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 9058–9067, 13–18 Jul 2020.

Wonchul Son, Jaemin Na, Junyong Choi, and Wonjun Hwang. Densely guided knowledge distillation using multiple teacher assistants. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9395–9404, October 2021.

Linh Tran, Bastiaan S. Veeling, Kevin Roth, Jakub Swiatkowski, Joshua V. Dillon, Jasper Snoek, Stephan Mandt, Tim Salimans, Sebastian Nowozin, and Rodolphe Jenatton. Hydra: Preserving ensemble diversity for model distillation. *CoRR*, abs/2001.04694, Mar. 2020.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H. Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 7611–7623. Curran Associates, Inc., Dec. 2020.

Blake Woodworth, Kumar Kshitij Patel, Sebastian Stich, Zhen Dai, Brian Bullins, Brendan Mcmahan, Ohad Shamir, and Nathan Srebro. Is local SGD better than minibatch SGD? In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 10334–10343. PMLR, 13–18 Jul. 2020.

Zhaohui Yang, Mingzhe Chen, Walid Saad, Choong Seon Hong, and Mohammad Shikh-Bahaei. Energy efficient federated learning over wireless communication networks. *IEEE Transactions on Wireless Communications*, 20(3):1935–1949, 2021.

Mikhail Yurochkin, Mayank Agarwal, Soumya Ghosh, Kristjan Greenewald, Nghia Hoang, and Yasaman Khazaeni. Bayesian nonparametric federated learning of neural networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 7252–7261. PMLR, 09–15 Jun. 2019. URL `https://proceedings.mlr.press/v97/yurochkin19a.html`.

Hailin Zhang, Defang Chen, and Can Wang. Confidence-aware multi-teacher knowledge distillation. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4498–4502. IEEE, 2022.

Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. Federated Learning with non-IID Data, Jun. 2018.

Xiatian Zhu, Shaogang Gong, et al. Knowledge distillation by on-the-fly native ensemble. *Advances in neural information processing systems*, 31, 2018.

Yichen Zhu and Yi Wang. Student customized knowledge distillation: Bridging the gap between student and teacher. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5057–5066, October 2021.

Zhuangdi Zhu, Junyuan Hong, and Jiayu Zhou. Data-free knowledge distillation for heterogeneous federated learning. In *International Conference on Machine Learning*, pp. 12878–12889. PMLR, 2021.

A BACKGROUND

A.1 STOCHASTIC GRADIENT DESCENT AND FEDERATED LEARNING

Consider an optimization on a Deep Neural Network (DNN) with the SGD algorithm and a set of parameters $\boldsymbol{\omega} = \{\omega_1, \omega_2, \ldots, \omega_p\}$ with p being the size of the DNN's model. SGD Robbins & Monro (1951) usually uses a mini-batch gradient Woodworth et al. (2020) $\widetilde{g}(\boldsymbol{\omega}) = -\nabla_{\boldsymbol{\omega}} g_b(\boldsymbol{\omega})$ instead of a full-batch gradient $g(\boldsymbol{\omega}) = -\nabla_{\boldsymbol{\omega}} f(\boldsymbol{\omega})$. The subscript b denotes the mini-batch index set, which is drawn from B batches randomly. Therefore, in a mini-batch SGD, the full-batch dataset $\mathcal{D}$ is divided into B mini batches: $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_B\}$. Thus, the mini-batch SGD satisfies:

$$g(\boldsymbol{\omega}) = \frac{1}{B} \sum_{b=1}^{B} \widetilde{g}(\boldsymbol{\omega}). \tag{13}$$

FL utilizes the concept of SGD. In terms of implicit optimization for FL, clients collect data in the areas which are under their supervision. Thus, data, collected by FL clients from each communication round, can act as a mini-batch of the SGD (the whole data in the FL system counts as the full-batch data) and contribute to the local training process on distributed clients. On completion of the local training at clients, the clients then send their local trained loss functions to the server for aggregation process McMahan et al. (2017). The original idea of implicit optimization in FL is similar to the relationship between the full-batch and mini-batch gradients in equation 13, which is as follows:

$$f(\boldsymbol{\omega}) = \frac{1}{N} \sum_{n=1}^{N} f_n(\boldsymbol{\omega}), \tag{14}$$

where $f(\boldsymbol{\omega})$ is the aggregated loss function at the server, and $f_n(\boldsymbol{\omega})$ is the resulting loss at the client n in the system with N clients. In order to simplify the FL process, the authors in McMahan et al. (2017) proposed the surrogate function, namely FedAvg, in which the generality of the FL process is preserved:

$$\widetilde{\boldsymbol{\omega}} = \frac{1}{N} \sum_{n=1}^{N} \boldsymbol{\omega}_n. \tag{15}$$

We have $\widetilde{\boldsymbol{\omega}}$ to be the global parameter set that is aggregated from the set of N clients in the system. Due to the aforementioned similarities between SGD and FL, FL possesses all characteristics of the SGD.

A.2 KNOWLEDGE DISTILLATION

Knowledge Distillation (KD) Hinton et al. (2015) employs a technique to transfer the learned knowledge from a pre-trained teacher model to another model with less or similar complexity (student model). The model transfer process is implemented in two steps. In the first step, a surrogate output probability function, namely a temperature-softmax function, is utilized. By adding a temperature scaling variable T, the conventional softmax function then becomes:

$$\hat{p}(l|\boldsymbol{X}, \boldsymbol{\omega}, T) = \frac{\exp(z^l(\boldsymbol{X})/T)}{\sum_{j=1}^{C} \exp(z^j(\boldsymbol{X})/T)}, \tag{16}$$

where z^j is the output set corresponding to class j of the given DNN with batch of input data $\boldsymbol{X}$. The subscript l denotes the index of softmax output which corresponds to the prediction on class l of the DNN. The intention of adding the variable T is to adjust the slope of the softmax function in the classifier as shown in equation 16. As we can see, when we increase the value of T, the slope of the softmax function will decrease significantly. With large temperature scale values, over the same output range of data, the range of values represented by input z^j is larger. Then, the output value tuple created by the teacher and student carries considerably more information. Therefore, the learning process between teacher and student is more effective.

In the second step, to help implement the transfer of knowledge from teacher to student, the authors in Hinton et al. (2015) presented a new loss function, called distillation loss function (also known as soft-loss function). This loss function comprises two terms: the intrinsic entropy function of the teacher and the cross-entropy function between the teacher and the student's outputs, which can be expressed as follows:

$$\mathcal{L}_{\text{KD}} = H(p(\boldsymbol{X}), q(\boldsymbol{X})) - H(p(\boldsymbol{X})) = \sum_{x_i \in \boldsymbol{X}} p(x_i) \log p(x_i) - \sum_{x_i \in \boldsymbol{X}} p(x_i) \log q(x_i). \tag{17}$$

The purpose of this function is to compare the output distribution between teacher $p(\boldsymbol{X})$ and student model $q(\boldsymbol{X})$. In terms of information theory, this measurement shows the under-performance of the distribution set created by student, when the output distribution set of the teacher is taken as the sample distribution set, with precision of 100%. By minimizing this function, we reduce the functional difference between the two deep networks. As a result, the student model tends to make it's output distribution become more identical to the teacher model's behavior.

Moreover, to restrain the student model from learning the teacher's false result, the authors in Hinton et al. (2015) proposed the hard-loss formula:

$$\mathcal{L}_{\text{CE}} = H(P, \hat{P}) = - \sum_{x_i \in \boldsymbol{X}} p(x_i) \log \hat{p}(x_i). \tag{18}$$

The hard-loss formula acts as a backing force with two main objectives. First of all, it ensures that the joint distillation function always follows the right optimizing track. Secondly, due to the highly complicated function made by the soft-loss function, the hard-loss force aids the model to escape the local minima whenever the model gets trapped. In this way, student's training performance improves significantly.

B ISSUES ABOUT SGD AND FEDERATED LEARNING

As mentioned in Appendix A.1, FL and SGD have the same attributes. Occasionally, SGD is an efficient method for DL. For a large dataset, SGD can converge faster as it causes updates to the parameters more frequently. Moreover, the steps taken towards the minima of the loss function have oscillations Shwartz-Ziv & Tishby (2017) that can help to get out of the local minima of the loss function Neelakantan et al. (2015); Smith et al. (2020); He et al. (2019). Therefore, FL can operate well under the ideal conditions. However, when the problems, such as heterogeneity on data and high complexity on training models come about, both SGD and FL face many obstacles.

B.1 THE GENERALIZATION GAP IN SGD AND FL

By measuring the distance between initial model and trained one, the research in Liu et al. (2020b) manifests the proof of low distance traveled by the high complexity model using SGD. The authors observed that Vanilla SGD suffers from high overfitting compared to other training optimizing methods, such as data augmentation, $L2$-regularization, and momentum. Moreover, the model's travelled distance which is trained by Vanilla SGD is 4 to 5 times lower than that is trained by the other methods. The research in Keskar et al. (2016) shows that the lack of generalization ability is because large-batch methods tend to converge to sharp minimizers of the training function. These minimizers are characterized by a significant number of large positive eigenvalues in Hessian matrix of the loss function $f(\boldsymbol{\omega})$, and tend to generalize less effectively. When considering each distributed node in the FL system as a mini-batch of SGD, we can imagine the current FL system as a large-batch for SGD, which can easily be trapped in the sharp minimum. For a more mathematical analysis, we consider the gradient dissimilarity inequality for the aggregated global model $\boldsymbol{\omega}$, there exist constants $G \geq 0$ and $U \geq 1$ that attain:

$$\frac{1}{N}\sum_{n=1}^{N}\|\nabla f_n(\boldsymbol{\omega})\|^2 \leq G^2 + U^2\|\nabla f(\boldsymbol{\omega})\|^2,\ \forall \boldsymbol{\omega}. \tag{19}$$

Here, we follow the heterogeneity assumption that G is defined with gradient variance boundary (on client n) σ_n^2 as in Mishchenko et al. (2019). Along with the condition that the problem $f_i(\boldsymbol{\omega})$ is μ-smooth, the assumption can be relaxed as in Karimireddy et al. (2020):

$$\frac{1}{N}\sum_{i=1}^{N}\|\nabla f_i(\boldsymbol{\omega})\|^2 - G^2 \approx \underbrace{\frac{1}{N}\sum_{i=1}^{N}\|\nabla f_i(\boldsymbol{\omega})\|^2}_{\mathbb{A}} - \underbrace{\frac{2}{N}\sum_{i=1}^{N}\|\nabla f_i(\boldsymbol{\omega}^*)\|^2}_{\mathbb{B}} \leq 2\mu B^2(f(\boldsymbol{\omega}) - f^*), \tag{20}$$

where f^* denotes the optimality. From this inequality, we can see that the distance between the current state $f(\boldsymbol{\omega})$ and the global optima state f^* is lower-bounded by $\mathbb{A} + \mathbb{B}$. As we can see from equation 20, the first term $\mathbb{A} = \frac{1}{N}\sum_{i=1}^{N}\|\nabla f_i(\boldsymbol{\omega})\|^2$ reveals the average gradient norm at the model state $\boldsymbol{\omega}$. When this term of value is large, the current state is assumed to have a high chance to be far from the true optimal state f^* (due to the term $\mathbb{B}$ being fixed). Simultaneously, the current state $\boldsymbol{\omega}$ is more likely to be a sharp minimizer because of a larger $\mathbb{A}$ value. The illustrative explanation about sharp and flat minimizers is presented in Fig. 4.

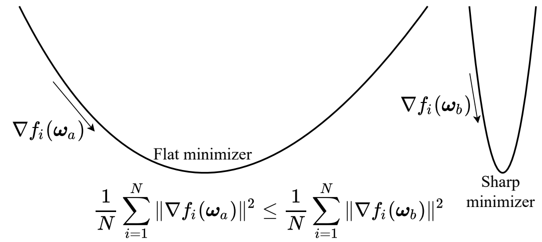

$\nabla f_i(\boldsymbol{\omega}_a)$
Flat minimizer
$\nabla f_i(\boldsymbol{\omega}_b)$
Sharp minimizer
$\frac{1}{N}\sum_{i=1}^{N}\|\nabla f_i(\boldsymbol{\omega}_a)\|^2 \leq \frac{1}{N}\sum_{i=1}^{N}\|\nabla f_i(\boldsymbol{\omega}_b)\|^2$

Figure 4: Illustration of flat and sharp minima.

B.2 CLIENT-DRIFT ON FL

The authors in Zhao et al. (2018) introduced a term named "weight divergence", which can be computed as $\|\boldsymbol{\omega}^{\mathbb{G}} - \boldsymbol{\omega}^{\mathbb{C}}\|$. This weight divergence is the total distance between two models: practical aggregated global model of the FL system $\boldsymbol{\omega}^{\mathbb{G}}$ and ideal model when trained with the data collected and shuffled one place $\boldsymbol{\omega}^{\mathbb{C}}$. For a mathematical explanation, weight divergence at epoch E of communication round m can be demonstrated as follows:

$$\begin{aligned}\|\boldsymbol{\omega}^{\mathbb{G}}_{mE} - \boldsymbol{\omega}^{\mathbb{C}}_{mE}\| \leq& \sum_{n=1}^{N} \frac{s^n}{\sum_{n=1}^{N} s^n} (a^n)^E \|\boldsymbol{\omega}^{\mathbb{G}}_{(m-1)E} - \boldsymbol{\omega}^{\mathbb{C}}_{(m-1)E}\| \\ &+ \eta \sum_{n=1}^{N} \frac{s^n}{\sum_{n=1}^{N} s^n} \sum_{c=1}^{C} \|p^n(y=c) - p(y=c)\| \times \sum_{e=0}^{E-1} (a^n)^e g_{\max}(\boldsymbol{\omega}^{\mathbb{C}}_{mE-1-n}),\end{aligned} \tag{21}$$

where η is the learning rate, s^n is the number of samples on client n in a set of N clients, $p^n(y=c)$ is the probability density of the class c on client n, and $p(y=c)$ is the global model's probability density. $g_{\max}(\boldsymbol{\omega}^{t}_{mE-1-n})$ is the max gradient over C classes, and a^n is the average smooth coefficient over C classes which is defined as: $1 + \eta \sum_{c=1}^{C} p^n(y=c)\mu_c$ with assumption that the problem $f_i(\boldsymbol{\omega})$ is μ_c-smooth on class c. As mentioned in Zhao et al. (2018), we have three following observations:

- The weight divergence is affected by two aspects: the weight divergence value from the last global aggregation, and the relative entropy between data distribution on client n and the actual distribution on the whole population.
- The term $\sum_{n=1}^{N} \frac{s^n}{\sum_{n=1}^{N} s^n} (a^n)^E$ acts as an amplifier. Regardless of the data property, the weight divergence still occurs to the aforementioned amplifier term. Therefore, the FL system always suffers the reduction in accuracy even if the data is IID.
- When all clients start from the same initialization, the probability distance $\|p^n(y=c) - p(y=c)\|$ is the matter of divergence. The illustration of the probability distance is described in Fig. 5

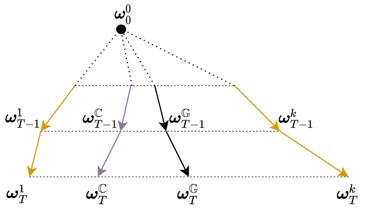

$\boldsymbol{\omega}_0^0$
$\boldsymbol{\omega}_{T-1}^1$
$\boldsymbol{\omega}_{T-1}^{\mathbb{C}}$
$\boldsymbol{\omega}_{T-1}^{\mathbb{G}}$
$\boldsymbol{\omega}_{T-1}^k$
$\boldsymbol{\omega}_T^1$
$\boldsymbol{\omega}_T^{\mathbb{C}}$
$\boldsymbol{\omega}_T^{\mathbb{G}}$
$\boldsymbol{\omega}_T^k$

Figure 5: Illustration of weight divergence in non-IID data.

C SYSTEM MODEL

As shown in Fig. 1, we first compartmentalize the network into R, where each region contains one access point (AP) acting as a local server (denoted by $r \in \{1, 2, \ldots, R\}$) and N clients (denoted by $n \in \{1, 2, \ldots, N\}$) are randomly sampled from the FL system. For each communication round of FL, those APs execute the FedAvg, which aggregates the model parameters from local predictors in their own regions, resulting in the regional aggregated model $\boldsymbol{\omega}^r$. The global server receives the aggregated model parameters after a certain number of communication rounds. Each updated model, thus, acts as a teacher to share knowledge to the global deep model $\boldsymbol{\omega}^g$. The data pool $\mathcal{S} = (\boldsymbol{X}, \boldsymbol{Y}) = \{(x_i, y_i)\}$ in which $i \in \{1, 2, \ldots, S\}$ is stored in the global server for the multi-teacher distillation process. Here, $y_i \in \{1, 2, \ldots, C\}$ is the truth label of the preset dataset $\mathcal{S}$. We define l as the output index of the deep model and $c \in \{1, 2, \ldots, C\}$ as the predicted label index of the data $\boldsymbol{X}$ when it is processed by the deep model.

D F2L ALGORITHM DETAILS

Algorithm 2 Label-driven Joint KD

```
Input: $\boldsymbol{\omega} = (\boldsymbol{\omega}^1, \boldsymbol{\omega}^2, \ldots, \boldsymbol{\omega}^R), \boldsymbol{\omega}^g$, $(\boldsymbol{X}, \boldsymbol{Y})$
initialize $\mathcal{L}_{\text{LKD}} = 0$
for each epoch e = 1,2,..., E do
    for each region r = 1,2,..., R do
        $\boldsymbol{\beta}_r \leftarrow$ C-Reliability$(\boldsymbol{\omega}^r, \boldsymbol{X})$ in Algorithm 6
        $\boldsymbol{X}^r_{\text{alg}}, \boldsymbol{Y}^r_{\text{alg}} \leftarrow$ L-SampleAlign$(\boldsymbol{X}, r, \boldsymbol{\omega}^r)$ in Algorithm 3
        $\mathcal{L}_{\text{LKD}} \leftarrow$ L-KD$(\boldsymbol{\omega}^r, \boldsymbol{X}^r_{alg}, \boldsymbol{Y}^r_{alg}, \boldsymbol{\beta}_r)$ in Algorithm 4
    end for
    $\boldsymbol{X}^g_{\text{alg}}, \boldsymbol{Y}^g_{\text{alg}} \leftarrow$ L-SampleAlign$(\boldsymbol{X}, r, \boldsymbol{\omega}^g)$
    $\mathcal{L}_{\text{LKD}} \leftarrow \mathcal{L}_{\text{LKD}} +$ G-update-KD$(\boldsymbol{\omega}, \boldsymbol{X}^g_{\text{alg}}, \boldsymbol{Y}^g_{\text{alg}})$
    $\mathcal{L}_{\text{LKD}} \leftarrow \mathcal{L}_{\text{LKD}} + \mathcal{L}_{\text{CE}}(\boldsymbol{\omega}^g, \boldsymbol{X}, \boldsymbol{Y})$
    Update the global parameters:
    $\boldsymbol{\omega}^{\boldsymbol{g}} \leftarrow \boldsymbol{\omega}^{\boldsymbol{g}} - \eta \nabla \mathcal{L}_{\text{LKD}}$ with $\eta$ is the learning rate
end for
```

Algorithm 3 L-SampleAlign

```
Input: $\boldsymbol{X}$, $r$, $\boldsymbol{\omega}^r$
for $x_i$ in $\boldsymbol{X}$ do
    $c \leftarrow$ Predict label on model $\boldsymbol{\omega}^r$
    $\boldsymbol{X}^r_{alg}[c], \boldsymbol{Y}^r_{alg}[c] \leftarrow x_i, y_i$
end for
```

Algorithm 4 L-KD

```
Input: $\boldsymbol{\omega}^r, \boldsymbol{X}^r_{alg}, \boldsymbol{Y}^r_{alg}, \boldsymbol{\beta}$
Initialize: $\mathcal{L}_r = 0$
for $c$ in $(1, 2, \ldots, C))$ do
    for $x_i$ in $\boldsymbol{X}^r_{\text{alg}}[c]$ do
        $\boldsymbol{p}_i \leftarrow$ Predict logit output on model $\boldsymbol{\omega}^r$
        $\boldsymbol{q}_i \leftarrow$ Predict logit output on model $\boldsymbol{\omega}^g$
        for $l$ in $(1, 2, \ldots, C)$ do
            $KL_l = p^l_i \times (\log p^l_i - \log q^l_i)$
            $\mathcal{L}_r \leftarrow \mathcal{L}_r + KL_l$
        end for
    end for
end for
```

Algorithm 5 G-Update-KD

```
Input: $\boldsymbol{\omega}, \boldsymbol{X}, \boldsymbol{Y}$
for $x_i$ in $\boldsymbol{X}^g_{\text{alg}}$ do
    for $l$ in $(1, 2, \ldots, C)$ do
        $p^l_i = q^l_i$
        $\boldsymbol{q}_i \leftarrow$ Predict logit output on model $\boldsymbol{\omega}^g$
        $KL_{\text{upd}} = p^l_i \times (\log p^l_i - \log q^l_i)$
        $\mathcal{L} = \mathcal{L} + KL_{\text{upd}}$
    end for
end for
```

Algorithm 6 C-Reliability

```
Input: $\boldsymbol{\omega}^r, \boldsymbol{X}$
Initialize:
for $r$ in $1, 2, \ldots, R$ do
    $S^l_r = 0$
end for
for $r$ in $1, 2, \ldots, R$ do
    $f^r_{\text{AUC}} =$ evaluate AUC on model $\boldsymbol{\omega}^r$
    for $l$ in $1, 2, \ldots, C$ do
        $S^l_r \leftarrow S^l_r + \exp(f^{l,r}_{\text{AUC}})$ with $f^{l,r}_{\text{AUC}} \in f^r_{\text{AUC}}$
    end for
end for
for $l$ in $1, 2, \ldots, C$ do
    $\beta^c_r \leftarrow \exp(f^{l,r}_{\text{AUC}})/S^l_r$
end for
```

E EXPERIMENT SETUP

Dataset. We evaluate all algorithms on non-IID data partitioning with Dirichlet random distribution function Li et al. (2022). We use benchmark datasets with the same train/test splits as in McMahan et al. (2017). We use four multi-class categorisation benchmark datasets in our evaluations, including **1)** MNIST LeCun & Cortes (2010), **2)** EMNIST Cohen et al. (2017), **3)** CIFAR-10 Krizhevsky (2009), **4)** CIFAR-100 Krizhevsky (2009), **5)** CINIC-10 Darlow et al. (2018), and **6)** CelebA gender classification Liu et al. (2015) . The evaluation settings for simulations are listed in Appendix M.

Settings. We apply two DNNs for our experimental evaluations, including LeNet-5 LeCun et al. (1989) for MNIST and EMNIST datasets, and ResNet-18 He et al. (2016) for CIFAR-10 and CIFAR-100. It is worth noting that we do not apply pre-trained models on both LeNet-5 and ResNet-18 because the pre-trained models can help FL approach the global minima easily, which can reduce the generalization of the evaluation. The detailed settings is defined in Appendix M

Baselines. We compare the performance on non-IID dataset with four baselines, including FedProx, FedDistill Chen & Chao (2020), FedGen Zhu et al. (2021), and FedAvg.

Evaluation metric. We adopt top-1 accuracy on every experiments to evaluate the performance of our F2L method and the other FL baselines. Further, each evaluation is an average of five results.

Settings for Scalability test. We use CINIC-10 dataset, where the data consists of 270.000 images (90.000 images in each train, validation, test set). We use the train set for the FL training, validation set for the accuracy evaluation. To make the dataset for the additional clients, we use the test set. The more detailed of the settings is demonstrated in Table 3. As described in Table 3, the data on every client is sampled with non-IID coeff $\alpha = 0.1$ (on both old and new regions). To make the test between FedAvg and F2L fair, we use the same number of users for added regions (i.e., total added clients is 100).

Table 3: The data settings used for scalability test.

Data Setting			
	Old Regions	Added Regions	Evaluation dataset
Data size	90000	90000	90000
non-IID coeff α	1	0.1	N/A
Vanilla FL system			
Clients per region	100	100	1
Samples per client	5000	5000	10000
F2L system			
Clients per region	$33, 33, 34$	$33, 33, 34$	1
Samples per client	5000	5000	10000
Samples on server	9000		

F PROOF ON RANDOM SAMPLING EFFECTS TO DATA DISTRIBUTION

Assumption 2 *We consider the class-likelihood of an data as the classical Gaussian Mixture Model (GMM). Therefore, the feature distribution on data can be represented as the Dirichlet Process GMM (DPGMM) DelPozzo et al. (2018).*

$$p\left(x|(\mu_c, \Sigma_c, \pi_c)_{c=1}^{C}\right) = \sum_{c=1}^{C} \pi_c \, \mathcal{N}(\mathbf{x}; \mu_c, \Sigma_c), \tag{22}$$

where π_c is the Dirichlet allocation weight on class c.

Lemma 3 *The data distribution on regional FL is a combination of client set, and can be represented as a DPGMM.*

$$p\left(x|(\mu_c, \Sigma_c, \pi_c)_{c=1}^{C}\right) = \sum_{c=1}^{C} \alpha_c \, \mathcal{N}(\mathbf{x}; \mu_c, \Sigma_c), \tag{23}$$

where α_c is the Dirichlet allocation weight on class c of data point j over total S_c^r data sampled from region r.

Proof: The proof is demonstrate in Appendix L.

To theoretically analyze the random sampling effects to regional data distribution, we need to prove that $\pi_c^r \neq \pi_{c'}^r$ and $\pi_c^r \neq \pi_c^{r'}$. To this end, we consider covariance $\mathrm{Cov}(\pi_c^r, \pi_c^{r'})$ and $\mathrm{Cov}(\pi_c^r, \pi_{c'}^r)$. Applying covariance equation for Dirichlet distribution MacKay (2003), we have:

$$\mathrm{Cov}(\pi_c^r, \pi_{c'}^r) = -\frac{\nu_c^r \nu_{c'}^r}{\bar{\nu}^2(\bar{\nu}+1)}, \tag{24}$$

$$\mathrm{Cov}(\pi_c^r, \pi_c^{r'}) = -\frac{\nu_c^r \nu_c^{r'}}{\bar{\nu}^2(\bar{\nu}+1)}. \tag{25}$$

As we can see from the two mentioned equations, the covariances $\mathrm{Cov}(\pi_c^r, \pi_c^{r'})$ and $\mathrm{Cov}(\pi_c^r, \pi_{c'}^r)$ is always lower than 0. Furthermore, when a system is suffered more seriously from non-IID, the concentration parameter (i.e., ν_c^r) becomes higher Hsu et al. (2019). This phenomenon makes the random sampling capable of trigger high class-wise bias on regional FL. Therefore, the F2L can be more effective on heterogeneous network by applying LKD (which is more briefly described in 3.2).

G LABEL DRIVEN KNOWLEDGE DISTILLATION DECOMPOSITION

Recall the KL divergence function in 200 (2005), the KL divergence can be expressed as the average distance between teacher and student's probability among all input value $x_i \in \boldsymbol{X}$. Combining the aforementioned statement with equation 2, we can decompose the KL divergence function into multiple sub-components as:

$$\begin{aligned}\mathcal{L}_r^{KL} &= \sum_{l=1}^{C}\sum_{i=1}^{S} \hat{p}^r(l|x_i, \boldsymbol{\omega}^r, T, c) \log \frac{\hat{p}^r(l|x_i, \boldsymbol{\omega}^r, T, c)}{\hat{p}^g(l|x_i, \boldsymbol{\omega}^g, T, c)} \\ &= \sum_{l=1}^{C}\sum_{c=1}^{C}\sum_{i=1}^{S_c^r} \hat{p}^r(l|x_i, \boldsymbol{\omega}^r, T, c) \log \frac{\hat{p}^r(l|x_i, \boldsymbol{\omega}^r, T, c)}{\hat{p}^g(l|x_i, \boldsymbol{\omega}^g, T, c)} = \sum_{c=1}^{C} D_{KL}^c(\hat{p}^r||\hat{p}^g). \end{aligned} \tag{26}$$

Here, S is the number of samples of the fixed dataset $\mathcal{S}$ on the server. $(\boldsymbol{X}_{\text{alg}}^r, \boldsymbol{Y}_{\text{alg}}^r)$ is the dataset which is pseudo labeled and aligned by regional model r, and we have $(\boldsymbol{X}_{\text{alg}}^r[c], \boldsymbol{Y}_{\text{alg}}^r[c])$ as the set of data with size of S_c^r labeled by the model r as c.

Although the same preset dataset is utilized on every teacher model, the different pseudo labeling judgments from different teachers lead to the different dataset tuples. The process of identifying S_c^r is demonstrated in Algorithm 3. Because the regional models label on the same dataset S, we have $\sum_{c=1}^{C} S_c^r = S$ for all regional models. $D_{KL}^c(\hat{p}^r||\hat{p}^g)$ is the c label-driven KL divergence between model r and model g.

By dividing the dataset by class (regarding the teacher's label-driven prediction), we have the teacher's prediction as the pseudo label. Therefore, we can leverage the unlabeled dataset to train the LKD at the server. Moreover, with the decomposed equation equation 26, we can scale the point-wise distribution distance between two model r, and g. Therefore, the scale can be leveraged to judge the teacher model's label driven performance, and then, the good knowledge can be distilled. The surrogate MTKD function, so called LKD is demonstrated as follows:

$$\begin{aligned}\textbf{P3}: \min \mathcal{L}_m^{KL} &= \sum_{r=1}^{R}\sum_{c=1}^{C} \beta_r^c \sum_{i=1}^{S_c^r}\sum_{l=1}^{C} \hat{p}^r(l|x_i, \boldsymbol{\omega}^r, T, c) \log \frac{\hat{p}^r(l|x_i, \boldsymbol{\omega}^r, T, c)}{\hat{p}^g(l|x_i, \boldsymbol{\omega}^g, T, c)} \\ &= \sum_{r=1}^{R}\sum_{c=1}^{C} \beta_r^c D_{KL}^c(\hat{p}^r||\hat{p}^g). \end{aligned} \tag{27}$$

H PROOF ON THEOREM 1

We consider the sum

$$S = \sum_{j=1}^{n}\sum_{k=1}^{n} (e^{\tau_j^c} - e^{\tau_k^c})(\sigma_{j,c}^2 - \sigma_{k,c}^2). \tag{28}$$

We have $\sigma_{r,c}^2$ sequence is non-increasing. Furthermore, the $e^{\tau_r^c}$ sequence is non-decreasing due to the K and Jensen's inequality Durrett (2010). Therefore $(e^{\tau_j^c} - e^{\tau_k^c})(\sigma_{j,c}^2 - \sigma_{k,c}^2) \leq 0 \quad \forall j, k$, and thus $S \leq 0$. Thus, we have:

$$\frac{2R}{\sum_{r=1}^{R} e^{\tau_r^c}} \sum_{r=1}^{R} e^{\tau_j^c} \sigma_{r,c}^2 - \frac{2}{\sum_{r=1}^{R} e^{\tau_r^c}} \sum_{r=1}^{R} e^{\tau_r^c} \sum_{k=1}^{R} \sigma_{r,c}^2 \leq 0, \tag{29}$$

which also means:

$$\frac{2R}{\sum_{r=1}^{R} e^{\tau_r^c}} \sum_{r=1}^{R} e^{\tau_r^c} \sigma_{r,c}^2 \leq \frac{2}{\sum_{r=1}^{R} e^{\tau_r^c}} \sum_{r=1}^{R} e^{\tau_r^c} \sum_{r=1}^{R} \sigma_{r,c}^2. \tag{30}$$

From equation 30, we can have the following deduction to the relationship between student models distilled by LKD and MTKD:

$$\sigma^{*2}_{\mathrm{LKD,g,c}} = \frac{1}{\sum_{r=1}^{R} e^{\tau_r^c}} \sum_{r=1}^{R} e^{\tau_r^c} \sigma^2_{r,c} \leq (\frac{\sum_{r=1}^{n} e^{\tau_r^c}}{\sum_{r=1}^{R} e^{\tau_r^c}})(\frac{1}{R} \sum_{r=1}^{R} \sigma^2_{r,c}) = \frac{1}{R} \sum_{r=1}^{R} \sigma^2_{r,c} \overset{(a)}{=} \sigma^{*2}_{\mathrm{MTKD,g,c}}. \tag{31}$$

We can prove (a) based on Lemma 1 by setting the LKD allocation weight set $\beta_1^c = \beta_2^c = \cdots = \beta_R^c = 1$.

I PROOF ON THEOREM 2

We consider the sum

$$S = \sum_{j=1}^{n} \sum_{k=1}^{n} (e^{\tau_j^c} - e^{\tau_k^c})(|\mu_{j,c} - \bar{\mu}_c| - |\mu_{k,c} - \bar{\mu}_c|). \tag{32}$$

We have $(|\mu_{j,c} - \bar{\mu}_c| - |\mu_{k,c} - \bar{\mu}_c|)$ sequence is non-decreasing. Furthermore, the $e^{\tau_r^c}$ sequence is non-decreasing due to the K and Jensen's inequality Durrett (2010). Therefore $(e^{\tau_j^c} - e^{\tau_k^c})(\sigma^2_{j,c} - \sigma^2_{k,c}) \leq 0 \quad \forall j, k$, $S \leq 0$. Thus, we have:

$$\frac{2R}{\sum_{r=1}^{R} e^{\tau_r^c}} \sum_{r=1}^{R} e^{\tau_j^c} |\mu_{r,c} - \bar{\mu}_c| - \frac{2}{\sum_{r=1}^{R} e^{\tau_r^c}} \sum_{r=1}^{R} e^{\tau_r^c} \sum_{k=1}^{R} |\mu_{r,c} - \bar{\mu}_c| \leq 0, \tag{33}$$

which also means:

$$\frac{2R}{\sum_{r=1}^{R} e^{\tau_r^c}} \sum_{r=1}^{R} e^{\tau_r^c} |\mu_{r,c} - \bar{\mu}_c| \leq \frac{2}{\sum_{r=1}^{R} e^{\tau_r^c}} \sum_{r=1}^{R} e^{\tau_r^c} \sum_{r=1}^{R} |\mu_{r,c} - \bar{\mu}_c|. \tag{34}$$

From equation 34, we can have the following deduction to the relationship between student models distilled by LKD and MTKD:

$$\begin{aligned} |\mu^*_{\mathrm{LKD,g,c}} - \bar{\mu}_c| &= \frac{1}{\sum_{r=1}^{R} e^{\tau_r^c}} \sum_{r=1}^{R} e^{\tau_r^c} |\mu_{r,c} - \bar{\mu}_c| \leq (\frac{\sum_{r=1}^{n} e^{\tau_r^c}}{\sum_{r=1}^{R} e^{\tau_r^c}})(\frac{1}{R} \sum_{r=1}^{R} |\mu_{r,c} - \bar{\mu}_c| \\ &= \frac{1}{R} \sum_{r=1}^{R} |\mu_{r,c} - \bar{\mu}_c| \overset{(b)}{=} |\mu^*_{\mathrm{MTKD,g,c}} - \bar{\mu}_c|. \end{aligned} \tag{35}$$

We can prove (b) based on Lemma 1 by setting the LKD allocation weight set $\beta_1^c = \beta_2^c = \cdots = \beta_R^c = 1$.

J PROOF ON LEMMA 1

We consider the KL-divergence equation:

$$D_{\mathrm{KL}} = \sum_{i=1}^{S} p^r(x_i) \log \frac{p^r(x_i)}{p^g(x_i)}. \tag{36}$$

By taking the assumption that each label-driven posterior distribution follows normal distribution from assumption 2, we have the following:

$$\begin{aligned} D_{\mathrm{KL}} &= \sum_{c=1}^{C} \sum_{i=1}^{S_r^c} p^r(x_i|y=c) \log \frac{p^r(x_i|y=c)}{p^g(x_i|y=c)} \\ &= \sum_{c=1}^{C} \mathbb{E}_{p^r(x|y=c)} \Big[\frac{1}{2}\Big(\frac{(x_i - \mu_{g,c})^2}{\sigma^2_{g,c}} - \frac{(x_i - \mu_{r,c})^2}{\sigma^2_{r,c}}\Big) + \ln \frac{\sigma_{r,c}}{\sigma_{g,c}}\Big] \\ &= \sum_{c=1}^{C} \mathbb{E}_{p^r(x|y=c)} \Big[\frac{1}{2}\Big(\frac{x_i^2 - 2\mu_{g,c}x_i + \mu^2_{g,c}}{\sigma^2_{g,c}} - 1\Big) + \ln \frac{\sigma_{r,c}}{\sigma_{g,c}}\Big]. \end{aligned} \tag{37}$$

Applying additional raw moments of the normal distribution

$$X \sim \mathcal{N}(\mu, \sigma^2) \rightarrow \begin{cases} E[x] & = \mu, \\ E[x^2] & = \mu^2 + \sigma^2, \end{cases} \tag{38}$$

the KL distance in equation 37 becomes:

$$\begin{aligned} D_{\text{KL}} &= \sum_{c=1}^{C} \mathbb{E}_{p^r(x)} \Big[\frac{1}{2} \Big(\frac{x_i^2 - 2\mu_{g,c} x_i + \mu_{g,c}^2}{\sigma_{g,c}^2} - 1 \Big) - \ln \frac{\sigma_{r,c}^2}{\sigma_{g,c}^2} \Big] \\ &= \sum_{c=1}^{C} \Big[\frac{1}{2} \Big(\frac{\mu_{r,c}^2 + \sigma_{r,c}^2 - 2\mu_{g,c}\mu_{r,c} + \mu_{g,c}^2}{\sigma_{g,c}^2} - 1 \Big) - \ln \frac{\sigma_{r,c}^2}{\sigma_{g,c}^2} \Big] \\ &= \frac{1}{2} \sum_{c=1}^{C} \Big[\frac{(\mu_{r,c} - \mu_{g,c})^2}{\sigma_{g,c}^2} + \frac{\sigma_{r,c}^2}{\sigma_{g,c}^2} - 1 - \ln \frac{\sigma_{r,c}^2}{\sigma_{g,c}^2} \Big]. \end{aligned} \tag{39}$$

Combining equation 39 and LKD from equation 27, we have:

$$\begin{aligned} \mathcal{L}_m^{KL} &= \sum_{c=1}^{C} \sum_{r=1}^{R} \beta_r^c \Big[\frac{(\mu_{r,c} - \mu_{g,c})^2}{\sigma_{g,c}^2} + \frac{\sigma_{r,c}^2}{\sigma_{g,c}^2} - \ln \frac{\sigma_{r,c}^2}{\sigma_{g,c}^2} \Big] \\ &= \sum_{c=1}^{C} \frac{1}{\sum_{r=1}^{R} e^{\tau^c}} \sum_{r=1}^{R} e^{\tau_r^c} \Big[\frac{(\mu_{r,c} - \mu_{g,c})^2}{\sigma_{g,c}^2} + \frac{\sigma_{r,c}^2}{\sigma_{g,c}^2} - \ln \frac{\sigma_{r,c}^2}{\sigma_{g,c}^2} \Big] \qquad (40) \\ &= \sum_{c=1}^{C} \frac{1}{\sum_{r=1}^{R} e^{\tau^c}} \sum_{r=1}^{R} e^{\tau_r^c} \frac{(\mu_{r,c} - \mu_{g,c})^2}{\sigma_{g,c}^2} + \sum_{c=1}^{C} \frac{1}{\sum_{r=1}^{R} e^{\tau^c}} \sum_{r=1}^{R} e^{\tau_r^c} \Big[\frac{\sigma_{r,c}^2}{\sigma_{g,c}^2} - \ln \frac{\sigma_{r,c}^2}{\sigma_{g,c}^2} \Big] \qquad (41) \end{aligned}$$

Firstly, we take $\sum_{c=1}^{C} (\sum_{r=1}^{R} e^{\tau^c})^{-1} \sum_{r=1}^{R} e^{\tau_r^c} (\mu_{r,c} - \mu_{g,c})^2 (\sigma_{g,c}^2)^{-1}$ into consideration, we have:

$$D_{B,\text{KL}}^c = \sum_{r=1}^{R} e^{\tau_r^c} \frac{(\mu_{r,c} - \mu_{g,c})^2}{\sigma_{g,c}^2} \tag{42}$$

The optimal state of $\sum_{r=1}^{R} e^{\tau_r^c} ((\mu_{r,c} - \mu_{g,c})^2 / \sigma_{g,c}^2)$ is when the 1st derivative equal to 0. We have:

$$\nabla D_{B,\text{KL}}^c = 2 \sum_{r=1}^{R} e^{\tau_r^c} \frac{(\mu_{r,c} - \mu_{g,c})}{\sigma_{g,c}^2} = 0, \tag{43}$$

which also means:

$$\sum_{r=1}^{R} e^{\tau_r^c} \mu_{g,c}^* = \sum_{r=1}^{r} e^{\tau_r^c} \mu_{r,c} \Leftrightarrow \mu_{g,c}^* \sum_{r=1}^{R} e^{\tau_r^c} = \sum_{r=1}^{r} e^{\tau_r^c} \mu_{r,c} \Leftrightarrow \mu_{g,c}^* = \frac{\sum_{r=1}^{r} e^{\tau_r^c} \mu_{r,c}}{\sum_{r=1}^{R} e^{\tau_r^c}} \tag{44}$$

Taking $\sum_{c=1}^{C} (\sum_{r=1}^{R} e^{\tau^c})^{-1} \sum_{r=1}^{R} e^{\tau_r^c} \Big[(\sigma_{r,c}^2 / \sigma_{g,c}^2) - \ln (\sigma_{r,c}^2 / \sigma_{g,c}^2) \Big]$ is when the derivative of $D_{B,\text{KL}}^c$ into consideration, we have:

$$\mathcal{L}_m^{KL} = \sum_{c=1}^{C} \frac{1}{\sum_{r=1}^{R} e^{\tau^c}} \sum_{r=1}^{R} e^{\tau_r^c} \Big[\frac{\sigma_{r,c}^2}{\sigma_{g,c}^2} - \ln \frac{\sigma_{r,c}^2}{\sigma_{g,c}^2} \Big]. \tag{45}$$

We consider the label-driven KL divergence:

$$D_{\text{KL}}^c = \frac{1}{\sum_{r=1}^{R} e^{\tau^c}} \sum_{r=1}^{R} e^{\tau_r^c} \Big[\frac{\sigma_{r,c}^2}{\sigma_{g,c}^2} - \ln \frac{\sigma_{r,c}^2}{\sigma_{g,c}^2} \Big]. \tag{46}$$

We then find a close-form of D_{KL}^c as follows:

$$\begin{aligned} D_{\text{KL}}^{c*} &= \frac{1}{\sum_{r=1}^{R} e^{\tau^c}} \Big[\frac{\sum_{r=1}^{R} e^{\tau_r^c} \sigma_{r,c}^2}{\sigma_{g,c}^2} - \ln \prod_{r=1}^{R} \Big(\frac{\sigma_{r,c}^2}{\sigma_{g,c}^2} \Big)^{e^{\tau_r^c}} \Big] \\ &\overset{(a)}{\geq} \frac{1}{\sum_{r=1}^{R} e^{\tau^c}} \Big[\frac{\sum_{r=1}^{R} e^{\tau_r^c} \sigma_{r,c}^2}{\sigma_{g,c}^2} - \ln \Big(\frac{\sum_{r=1}^{R} e^{\tau_r^c} \sigma_{r,c}^2}{\sigma_{g,c}^2} \Big) \Big]. \end{aligned} \tag{47}$$

The inequality (a) holds due to the Young's inequality for products Bobkov et al. (2011). To understand the performance of LKD, we consider the optimal state of label-driven KL divergence (i.e., $D_{\mathrm{KL}}^{c*} = \min D_{\mathrm{KL}}^{c}$). Set $(\sum_{r=1}^{R} e^{\tau_r^c}\sigma_{r,c}^2)/(\sigma_{g,c}^*)^2 = u_r$. Then, we consider the function $f(x) = x - \ln x$. The optimal state of $f(x)$ is when derivative of $f(x)$ is 0, which also means: $\nabla f(x) = 0$. Therefore, we have:

$$\nabla f(x) = 1 - \frac{1}{x} = 0 \tag{48}$$

Therefore, the function $f(x)$ receives the optimal state when $x = 1$.

Taking the optimal result of equation 44 and equation 48, we have the followings:

$$\sigma_{\mathrm{LKD},g,c}^{*2} = \frac{1}{\sum_{r=1}^{R} e^{\tau^c}} \sum_{r=1}^{R} e^{\tau_r^c}\sigma_{r,c}^2, \tag{49}$$

$$\mu_{\mathrm{LKD},g,c}^{*} = \frac{1}{\sum_{r=1}^{R} e^{\tau^c}} \sum_{r=1}^{R} e^{\tau_r^c}\mu_{r,c}. \tag{50}$$

K PROOF ON LEMMA 2

Let b_c is the global optimal boundary to classify label c. We have the condition that the model r have the accurate prediction on data that have label c is $\mathcal{F}(x_i) \leq b_c$. Applying Central Limit Theorem (CLT) Durrett (2010), we have:

$$\tau_r^c = \Pr\Big[\mathcal{F}(\mathrm{x_i}) \leq \mathrm{b_c}\Big] \geq 1 - \frac{1}{\sqrt{2\pi}} \exp\Big[-\frac{1}{2}\Big(\frac{\mathrm{b_c}}{\sigma_{\mathrm{r,c}}}\Big)^2\Big]. \tag{51}$$

As we can see from equation 51, τ_r^c is proportional to $1/\sigma_{r,c}^2$. Thus we have when $\sigma_{1,c}^2 \leq \sigma_{2,c}^2 \leq \cdots \leq \sigma_{R,c}^2$, the model accuracy must satisfy the constraint $\tau_{1,c}^2 \geq \tau_{2,c}^2 \geq \cdots \geq \tau_{R,c}^2$.

L PROOF ON LEMMA 3

Due to the non-IID settings on FL, the data class can be distributed as Dirichlet process Yurochkin et al. (2019). To be more detailed, the data distribution can be represented as follows:

$$p\left(x|y=c\right) = \pi_j. \tag{52}$$

We have π_j is the Dirichlet allocation weight of data on ground truth with label l.

$$\pi_j^l = \frac{1}{\mathcal{B}(\nu)} \prod_{c=1}^{C} x_c^{\nu_c - 1} \quad \text{where} \quad \mathcal{B}(\nu) = \frac{\prod_c^C \Gamma(\nu)}{\Gamma\Big(\sum_{c=1}^{C} \nu_c\Big)}, \tag{53}$$

where $\Gamma(\nu)$ is the gamma function and ν is the Dirichlet coefficient. Applying equation 52 into Lemma 2, we have the following equation:

$$\begin{aligned} p\left(x|(\mu_c, \Sigma_c, \pi_c)_{c=1}^{C}\right) &= \sum_{c=1}^{C} \pi_c \sum_{j=1}^{S_c^r} \mathcal{N}(\mathbf{x}; \mu_c, \Sigma_c) && (54)\\ &= \sum_{c=1}^{C} \pi_c \pi_l S_c^r \, \mathcal{N}(\mathbf{x}; \mu_c, \Sigma_c) && (55)\\ &= \sum_{c=1}^{C} \alpha_c^r \, \mathcal{N}(\mathbf{x}; \mu_c, \Sigma_c). && (56) \end{aligned}$$

Here, we denote $\alpha_c^r = \pi_c \pi_l S_c^r$. Since π_c, π_l follow Dirichlet distribution, and S_c^r is a constant, $\pi_{c,j}$ follows Dirichlet distribution.

M DATA AND PARAMETER SETTINGS

Table 4: The data settings used for evaluation.

Data Setting						
	MNIST	EMNIST	CIFAR-10	CIFAR-100	CINIC-10	CelebA
Data size	48000	118440	48000	48000	48000	200000
Regions	3	3	3	3	3	3
non-IID coeff α	$0.1, 1$	$0.1, 1$	$0.1, 1$	$0.1, 1$	$0.1, 1$	$0.1, 1$
Federated Learning system						
Clients per region	10	10	10	10	10	10
Samples per client	1600	3948	1600	1600	1600	5000
Samples on server	3200	8000	3200	3200	3200	10000
Client settings						
Training models	LeNet-5	LeNet-5	ResNet-50	ResNet-50	ResNet-50	ResNet-50
Epochs	5	5	5	5	5	5
Learning rate	0.05	0.05	0.05	0.05	0.05	0.05

N EXPERIMENTAL RESULTS ON SOFT-LOSS CONTRIBUTION

Table 5: Top-1 accuracy of F2L on dataset EMNIST with different hard-loss scaling coefficients λ_3. The soft-loss λ_1 and λ_2 is scaled to be $1 - \lambda_3$ for a clear evaluation. The teacher efficiency is observed at the 5th distillation round (which is at every 40 rounds of communication).

	$\lambda_3 = 0$	$\lambda_3 = 0.001$	$\lambda_3 = 0.01$	$\lambda_3 = 0.1$	$\lambda_3 = 0.5$	$\lambda_3 = 1$
Teacher 1	64.33	64.33	64.33	64.33	64.33	64.33
Teacher 2	73.09	73.09	73.09	73.09	73.09	73.09
Teacher 3	74.66	74.66	74.66	74.66	74.66	74.66
G-student	71.75	82.96	**85.87**	**85.20**	84.30	82.29

Table 6: Top-1 accuracy of F2L on dataset CIFAR-10 with different hard-loss scaling coefficients λ_3. The soft-loss λ_1 and λ_2 is scaled to be $1 - \lambda_3$ for a clear evaluation. The teacher efficiency is observed at the 5th distillation round (which is at every 40 rounds of communication).

	$\lambda_3 = 0$	$\lambda_3 = 0.001$	$\lambda_3 = 0.01$	$\lambda_3 = 0.1$	$\lambda_3 = 0.5$	$\lambda_3 = 1$
Teacher 1	32.11	32.11	32.11	32.11	32.11	32.11
Teacher 2	30.64	30.64	30.64	30.64	30.64	30.64
Teacher 3	27.53	27.53	27.53	27.53	27.53	27.53
G-student	25.26	**41.32**	**52.71**	49.98	48.80	47.98

Table 7: Top-1 accuracy of F2L on dataset CIFAR-100 with different hard-loss scaling coefficients λ_3. The soft-loss λ_1 and λ_2 is scaled to be $1 - \lambda_3$ for a clear evaluation. The teacher efficiency is observed at the 5th distillation round (which is at every 40 rounds of communication).

	$\lambda_3 = 0$	$\lambda_3 = 0.001$	$\lambda_3 = 0.01$	$\lambda_3 = 0.1$	$\lambda_3 = 0.5$	$\lambda_3 = 1$
Teacher 1	6.62	6.62	6.62	6.62	6.62	6.62
Teacher 2	6.15	6.15	6.15	6.15	6.15	6.15
Teacher 3	7.56	7.56	7.56	7.56	7.56	7.56
G-student	7.40	**15.08**	**15.26**	**14.33**	11.44	9.64

O EXPERIMENTAL RESULTS ON REQUIRED TRAINING SAMPLE SIZE

Table 8: Top-1 accuracy of F2L on dataset EMNIST with different data-on-server numbers of samples. δ represents the sample scaling ratio compared to the data-on-server as demonstrated in Table 4. The teacher efficiency is observed at the 5^{th} distillation round (which is at every 40 rounds of communication).

	$\delta = 1$	$\delta = 1/2$	$\delta = 1/4$	$\delta = 1/6$	$\delta = 1/8$	$\delta = 1/10$
Teacher 1	64.33	64.33	64.33	64.33	64.33	64.33
Teacher 2	73.09	73.09	73.09	73.09	73.09	73.09
Teacher 3	74.66	74.66	74.66	74.66	74.66	74.66
G-student	**85.20**	**83.96**	80.71	79.48	76.64	74.22

Table 9: Top-1 accuracy of F2L on dataset CIFAR-10 with different data-on-server numbers of samples. δ represents the sample scaling ratio compared to the data-on-server as demonstrated in Table 4. The teacher efficiency is observed at the 5^{th} distillation round (which is at every 40 rounds of communication).

	$\delta = 1$	$\delta = 1/2$	$\delta = 1/4$	$\delta = 1/6$	$\delta = 1/8$	$\delta = 1/10$
Teacher 1	32.11	32.11	32.11	32.11	32.11	32.11
Teacher 2	30.64	30.64	30.64	30.64	30.64	30.64
Teacher 3	27.53	27.53	27.53	27.53	27.53	27.53
G-student	**54.71**	**52.04**	42.22	38.18	37.48	34.54

Table 10: Top-1 accuracy of F2L on dataset CIFAR-100 with different data-on-server numbers of samples. δ represents the sample scaling ratio compared to the data-on-server as demonstrated in Table 4. The teacher efficiency is observed at the 5^{th} distillation round (which is at every 40 rounds of communication).

	$\delta = 1$	$\delta = 1/2$	$\delta = 1/4$	$\delta = 1/6$	$\delta = 1/8$	$\delta = 1/10$
Teacher 1	6.62	6.62	6.62	6.62	6.62	6.62
Teacher 2	6.15	6.15	6.15	6.15	6.15	6.15
Teacher 3	7.56	7.56	7.56	7.56	7.56	7.56
G-student	**15.41**	14.98	14.33	11.25	10.83	9.51