FedTA: Teacher Assistant Knowledge Distillation in non-IID Federated Learning

Rudra Barua
Harvard College
Cambridge, Massachusetts
rudrabarua@college.harvard.edu

Neeyanth Kopparapu
Harvard College
Cambridge, Massachusetts
nkopparapu@college.harvard.edu

Andrew Sima
Harvard College
Cambridge, Massachusetts
asima@college.harvard.edu

Jeremy Zhang
Harvard College
Cambridge, Massachusetts
jeremyzhang@college.harvard.edu

***Abstract*—Federated learning enables a global machine learning model to be trained without the data ever being shared with a central server or across devices. However, federated learning is limited by the on-device memory capacity and the communication costs with a global model. Furthermore, current federated learning systems have been shown to have significant drops in performance when client data distributions are no longer independently, identically, distributed. We developed Federated Learning with Teaching Assistants, FedTA, to decrease the communications costs through knowledge distillation-based model compression and improving accuracy on non-iid data through an intermediate layer of teaching assistants. This method has been effective in alleviating these bottlenecks and improving the performance gap for non-iid data.**

***Index Terms*—federated learning, knowledge distillation, non-iid data**

I. Introduction

Machine learning is used in a variety of applications across our daily lives: speech recognition, search engines, fraud detection, etc. Most of these machine learning algorithms involve building models based on large amounts of high-quality and diverse data to make these important day to day predictions or decisions. However, such large data sets are often proprietary or expensive. Furthermore, such data is even harder to access in fields such as healthcare where patient and clinical data are subject to strict privacy regulations. Intuitions using their own data could yield a biased model, but at the same time, sharing sensitive data across institutions would also pose privacy issues.

Federated learning is a machine learning technique that addresses these issues by bringing the model to the source of the data rather than the data being shared with the model [1]. This technique allows multiple nodes or devices to use their own local data to train the global model locally and send updates to a central model. Since the training is performed locally, the local data of each device is never shared with other devices or directly sent to a central server. Hence, this allows for a global model to be built without ever having direct access to local data, which is extremely helpful in applying machine learning to privacy-sensitive data.

However, many traditional federated learning methods face communication constraints and performance issues when dealing with non-independent, identically distributed (non-iid) data across devices. In real-life applications, federated learning requires recurrent communication across various devices. Enough bandwidth is needed exchange updates between the local device and central model. Furthermore, devices have a wide ranging amount of local memory and computing power. This severely limits the ability to train larger models. Finally, in practical applications such as medicine, data is rarely ever independent and identically distributed across devices. This ultimately leads to variations in when convergence is reached across devices during training. By using knowledge distillation and teaching assistants, we are hoping to reduce communication costs and time to convergence while maintaining or improving accuracy in federated learning.

II. Problem to Solve

A. Problem Statement and Background

Machine learning applied to the real world faces two major problems: the data is frequently non-iid and must be kept private due to the sensitive nature of the data. Federated learning attempts to alleviate the second issue of sharing data to other parties by making training local. However, devices are limited by on-device memory, high communication costs with the central model, and waiting for a certain number of devices need to be ready before updates can be pushed to a central server. Communication costs are expensive if the model is large, which limits the use of models like BERT [2]. Furthermore, knowledge distillation via teacher assistant has been shown to improve accuracy and reduce the gap between student and teacher models [3]. Hence, there is potential to improve federated learning by compressing models by teaching a smaller network to mimic a larger model through knowledge distillation [4].

In the case of non-iid data across devices, knowledge distillation between the central server and edge devices may not be enough to overcome the variance brought about by this non-iid data. An intermediate level of specialized teaching assistants for certain classes or distributions of data could ease the potential conflicts that updates from differently distributed data can send to the global models, which would decrease time to convergence and increase overall accuracy.

B. Project Goal

The goal of our project is maintain or improve the accuracy of federated learning training on non-iid data while reducing communication costs through an additional layer of teaching assistants combined with knowledge distillation.

C. Evaluation Metrics

To evaluate FedTA, we will be comparing its performance against other baseline models on non-iid CIFAR-10 data [5]. We are interested in looking at the classification accuracy, time until convergence, the overall trade off between using teachings assistants with model compression and accuracy.

III. Proposed Approach, Novelty, and Secret Weapon

In this paper, we propose Federated Teaching Assistant, or FedTA, a method that uses an intermediate level of teaching assistants to perform knowledge distillation for model compression and soften the distribution of the data to group similarly distributed data to certain teaching assistants. We want explore how we can use a teacher, teaching assistant, and student knowledge distillation model to maintain or improve accuracy while reducing communication costs in federated learning.

We believe that another layer of aggregation can help training converge for non-iid data. This is because knowledge distillation generates a softer distribution of probabilities for output classes. Over many iterations, specific teaching assistants would become specialized for specific distributions. Having specialized teaching assistants for certain distributions would improve the accuracy and performance of the global model.

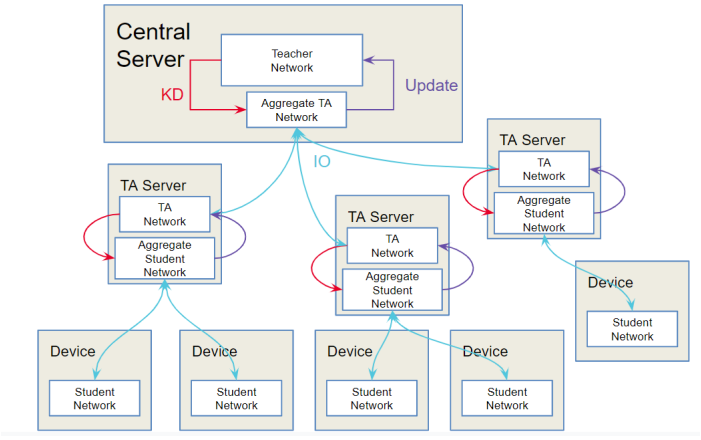

Fig. 1. FedTA Network Architecture

Algorithm 1 FedTA Algorithm

```
procedure FedTA_training(num_teacher_epochs)
    for ep < num_teacher_epochs do
        teacher_labels ← generate_labels(teacher)
        KD_train(teacher, teacher_labels)
        chosen_TAs ← random_choice(TAs)
        for TA in chosen_TAs do
            KD_train(TA, TA_labels)
            chosen_s ← random_choice(students)
            for student in chosen_s do
                for epoch < num_epochs do
                    train(student)
                    test(student)
                    epoch ← epoch +1
                end for
                agg_student ← average_weight(chosen_s)
                TA ← update(agg_student)
            end for
            agg_TA ← average_weight(chosen_TAs)
            teacher ← update(agg_TA)
            train(teacher)
            test(teacher)
            ep ← ep +1
        end for
```

IV. Intellectual Points

Models and updates need to be frequently exchanged between the global server and the devices in federated learning. Hence, the training process would be limited by the size of the models and bandwidth. We believe that the model compression that occurs through knowledge distillation will alleviate these I/O bottlenecks since communication would involve much smaller models than prior. Furthermore, knowledge distillation would better guarantee that the local model being trained meets the the memory and computation constraints of the local device.

In addition, FedTA offers accuracy improvements, due to how TA servers are able to mitigate non-iid data. Assuming some initial distribution of classes in the dataset, because of the large proportion of students, we expect by the law of large numbers that certain TAs will have a disproportionate amount of data of a certain class. It follows that these TA servers now have access to a dataset that is more iid than the original non-iid sample, which will surely offer accuracy gains post training.

V. Work Performed

A. Implementation

We first implemented an experimental framework that allowed us to simulated federated learning and the proposed FedTA architecture with one central teacher network, 20 TA servers, and 600 student devices. The training was then divided into three major steps: the initialization step, initial training step, and main FedTA training step.

In the initialization step, the teacher network is initialized with random weights and the data is split into public and private data for each device with non-iid sampling.

In the initial training step, the teacher model is trained for five epochs to create a starting baseline.

In the main FedTA training loop, teacher labels are first created for the data. We then implemented the architecture proposed in Figure 1. First, the aggregate TA network is trained on the new labels. After the training is complete, the aggregate TA model is given to all TA servers, where each TA then creates labels on the data that they are given. The aggregate student model on each TA then trains on these labels as well as some private data and distributes the trained aggregate student model to all of the student devices. On each student/local device, further training is done with local data. After local training, the weights for all of the students assigned to each TA are aggregated to create a new aggregate student model. The TA network on each TA server is then updated based on this new aggregate student model. Finally, all of the TA networks are used to create a new aggregate TA model on the central server, which then update the teacher network.

B. Theory

All theoretical work was based on randomization; by randomizing the assignment of student devices to teachers, we hoped to achieve improvements to accuracy. Suppose there are N student devices (essentially N samples from the dataset), with k TA servers that we intend to distribute these devices between. Post training there should exist N sets of weights, with W_m denoting the training results of student device m. If we assume that similar weights imply similar datasets, then intuitively we cluster the similar weights to obtain the results of datasets that are more iid.

Now assume there is some perfect function $\mathcal{S}$ comparing these sets of weights. Further denote the accuracy as a function of each of the average weights of the k TA servers as $\text{acc}(TA_1, TA_2, \ldots, TA_k)$. It follows that there should be some optimal partition to k groups given by $S_1, S_2, \ldots S_k$ such that we achieve the maximum theoretical accuracy:

$$TA_i = \frac{\sum_{W' \in S_i} W'}{|S_i|},$$
$$\text{max accuracy} = \max(\text{acc}(TA_1, TA_2, \ldots, TA_k)).$$

However, reaching this optimal point would require too much information across each of the student devices, defeating the privacy features offered by federated learning.

Instead, we rely on the law of large numbers and make the argument randomization will generate a disproportionate amount of data from some given class in certain TAs, resulting in a distribution across classes that is more similar to iid data. We further introduced knowledge distillation, as an attempt to utilize the dark knowledge between both teacher-TA and TA-student to dim the effects of the original non-iid data.

C. Data Used

For evaluating our approach against another baseline, we use the CIFAR-10 data set. This was the standard data set we came across in our federated learning research. For this project we were mainly interested in evaluating the performance of our model on non-iid data. Furthermore, since data is rarely ever independent and identically distributed in the real-world, we thought it was more important to compare FedTA to a baseline based on performance on non-iid data.

VI. Results and Discussion

A. Accuracy

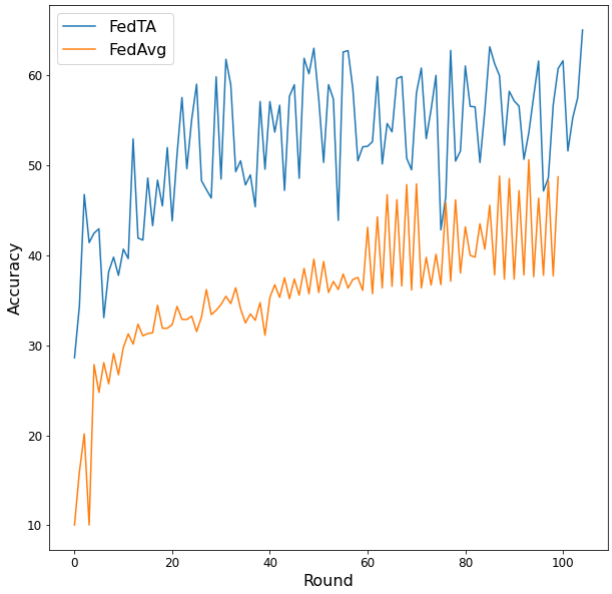

Fig. 2. Accuracy of FedAvg and FedTA

In Figure 2, we ran our FedTA algorithm for 100 rounds and plotted the test accuracy over these rounds. For comparison, we also plotted FedAvg test accuracy on the same graph. We see that our FedTA algorithm has higher variance in the test accuracy from round to round compared to FedAvg, but despite this variance, at each round, FedTA performed better than FedAvg and ultimately reached a test accuracy of 68.2%.

B. Effects of Quantization

We also explored the effects of quantization on the overall accuracy of our FedTA architecture. Looking at Figure 3, we see that overall, quantization does not negatively affect the accuracy significantly. On a more granular level, we see that as expected, 16-bit quantization reduces accuracy slightly, while 4-bit quantization further reduces the accuracy compared to using 32 bits. Considering the overall benefits of memory reduction and faster training times brought about with quantization, this may be a tradeoff that is worth making in the real world.

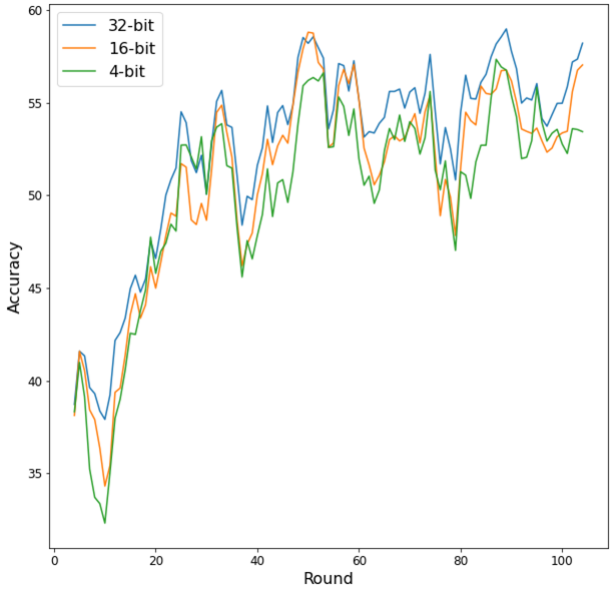

Fig. 3. Accuracy of FedTA With Quantization

C. Comparison to Related Work

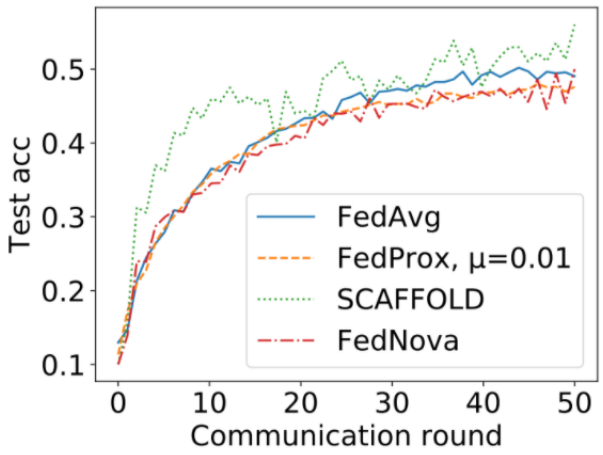

Fig. 4. Performance of Other Models

Based on this graph from existing literature comparing different federated learning architectures, we see that FedAvg has an accurate rate of approximately 49.8% while FedProx has an accuracy rate of approximately 50.7% [6]. Compared to the nearly 70% accuracy achieved by FedTA on the same CIFAR-10 dataset, we see that FedTA performs much better than the current literature.

VII. Conclusion

A. Contribution and Assessment

In this paper, we have introduced FedTA as a novel method for addressing non-iid federated learning while maintaining a high degree of accuracy without significant IO requirements. From our experimentation, we see that FedTA performs better than other federated learning architectures on similar datasets.

B. Future Work

Future work can be conducted to evaluate if the proposed architecture works on different datasets, such as CIFAR-100, or different types of data, such as MNIST. FedTA may also be an interesting method of mitigating malicious device attacks in federated learning. We believe that because devices can only connect to TA networks rather than the central server directly, it may be more difficult for an attacker to poison the central model. Further examination and experimentation will be needed to determine the feasibility and likelihood of attack.

C. Distribution of Work Performed

All authors participated and contributed equally throughout the project. More specifically, Rudra implemented many of the utility functions in the code and prepared the graphs for the report. Neeyanth implemented the main training loop and wrote the abstract and work performed sections. Andrew experimented with quantization and wrote about intellectual points and results. Jeremy proposed the initial architecture and idea, conducted experimentation, prepared diagrams and pseudocode for the report, as well as writing all of the remaining sections. We want to thank Professor H.T. Kung and the teaching staff for their generous guidance and support this semester.

References

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[3] S. I. Mirzadeh, M. Farajtabar, A. Li, N. Levine, A. Matsukawa, and H. Ghasemzadeh, "Improved knowledge distillation via teacher assistant," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, 2020, pp. 5191–5198.

[4] C. Wu, F. Wu, R. Liu, L. Lyu, Y. Huang, and X. Xie, "Fedkd: Communication efficient federated learning via knowledge distillation," *arXiv preprint arXiv:2108.13323*, 2021.

[5] A. Krizhevsky, V. Nair, and G. Hinton, "Cifar-10 (canadian institute for advanced research)." [Online]. Available: http://www.cs.toronto.edu/ kriz/cifar.html

[6] Q. Li, Y. Diao, Q. Chen, and B. He, "Federated learning on non-iid data silos: An experimental study," *arXiv preprint arXiv:2102.02079*, 2021.