A High-Efficiency Federated Learning Method Using Complementary Pruning for D2D Communication (Student Abstract)

Xiaoqing Xu[1], Jiaming Pei[2], Lukun Wang[1]

[1] School of Intelligent Equipment, Shandong University of Science and Technology, Tai'an, Shandong, China
[2] School of Computer Science, The University of Sydney, Camperdown, NSW, Australia
202383230053@sdust.edu.cn, jpei0906@uni.sydney.edu.au, wanglukun@sdust.edu.cn

Abstract

In federated learning, frequent parameter transmission between clients and the server results in significant communication overhead, particularly due to redundancy within the parameters. To address this issue, we propose a Complementary Pruning for Device-to-Device Communication (FedCPD) method. This approach effectively reduces the amount of transmitted parameters by applying complementary pruning techniques on both the server and clients. Additionally, we decrease the communication frequency between clients and the server by employing chain updates among clients (i.e., device-to-device communication). We conducted experiments on the MNIST, FMNIST, CIFAR-10, and CIFAR-100 datasets, and the results demonstrate that our method significantly reduces communication costs while improving model accuracy.

Introduction

Federated learning is a distributed machine learning approach that does not require centralizing data. Instead, it allows each device to train locally and transmit the updated parameters to the server. To address the high communication costs in federated learning, model pruning techniques reduce model complexity and the number of parameters by trimming smaller weights in the network, achieving sparsification. This approach not only effectively reduces the model size but also decreases the amount of data that needs to be transmitted.

Existing pruning methods can reduce the amount of transmitted data, but they often struggle to accurately identify which weights are critical to the model's performance. Consequently, during the pruning process, some weights that appear insignificant may be removed, even though they could be important in subsequent training or inference stages. This erroneous pruning weakens the model's expressive capacity, limiting its ability to capture complex features and ultimately affecting its performance.

Furthermore, many studies indicate that due to the differences in physical distance and network conditions between devices and the server (D2S), D2S often experiences high communication latency. By adopting direct communication between devices (D2D communication) to replace some of the communication between devices and the central server, reliance on the central server can be significantly reduced, thereby enhancing communication efficiency.

Thus, this paper proposes the FedCPD algorithm, which combines complementary pruning with device-to-device (D2D) communication. Unlike traditional methods, FedCPD not only retains the updated data for pruned parameter locations but also employs a chain update method that allows each client to sequentially receive and utilize the updated parameters from the previous client. This approach significantly reduces the frequency of communication between the server and clients while maintaining or even improving model performance. Compared to traditional methods, FedCPD effectively lowers communication overhead and enhances training efficiency, optimizing the overall performance of federated learning.

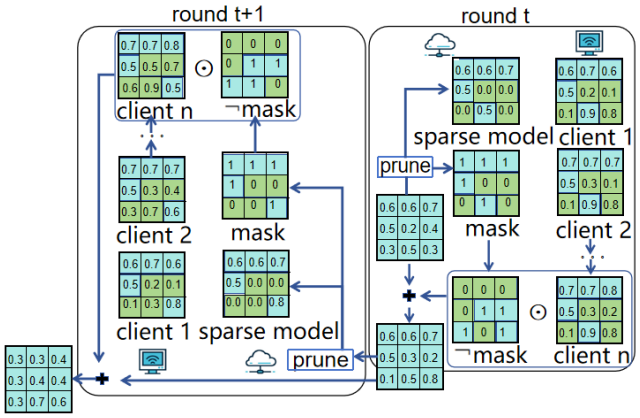


Figure 1: The training process flowchart of the FedCPD algorithm from round t to round $t+1$.

Methodology

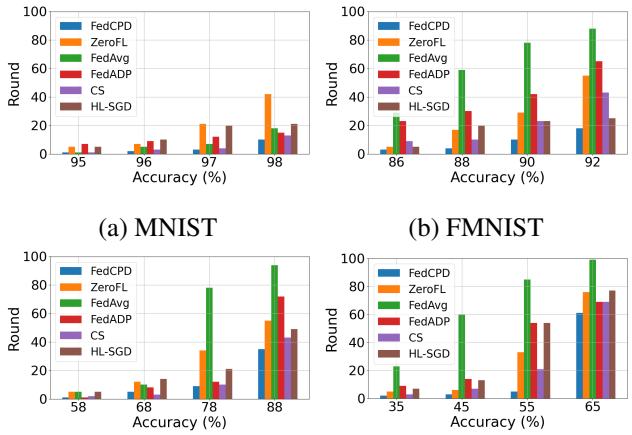
The training process of the FedCPD method from round t to round $t+1$ is shown in Figure 1. This method uses pruning masks to track the pruned parameters and reduces dependency on the server by employing client chain updates.Only the first client needs to download the model parameters from the server, while intermediate clients use the parameters updated by the previous client for local training. Once the

MNSIT				FMNIST		
Round	Accuracy(%)	Communication overhead (/Mb)	Flops (TFLOPs)	Accuracy(%)	Communication overhead (/Mb)	Flops (TFLOPs)
10	93	0.3E+05	0.4	84	0.8E+07	0.5
30	97	0.8E+06	0.7	90	1.3E+04	0.9
90	98	1.6E+07	1.2	93	2.4E+03	1.4
CIFAR-10				**CIFAR-100**		
Round	Accuracy(%)	Communication overhead (/Mb)	Flops (TFLOPs)	Accuracy(%)	Communication overhead (/Mb)	Flops (TFLOPs)
10	56	1.1E+04	2.1	36	2.5E+02	2.4
30	78	2.2E+08	3.0	62	4.1E+06	3.4
90	91	5.3E+03	4.9	79	7.8E+07	6.2

Table 1: The training results of the FedCPD algorithm on four datasets.

last client completes its training, the final parameters are multiplied by the inverse pruning mask using a Hadamard product, then uploaded to the server, where they are aggregated with the pre-pruning global model to obtain a new global model. The method ensures high accuracy while reducing communication overhead by combining complementary pruning with device-to-device (D2D) chain training.

Here, w_t represents the model weights at round t, and θ_t denotes the local model at the same round. During the training process, the server first performs L1 pruning on the global model to create a sparse version and records the corresponding pruning mask as mask. The first client receives this sparse model, updates its local model θ_{t+1} after training, and then passes it to the next client for chain updates. After the last client finishes its training, the final model parameters $\theta_{t+1,n}$ are multiplied by the inverse pruning mask using a Hadamard product before being uploaded to the server. The server then aggregates these parameters with the initial sparse model, filling in the missing parameters, and continues the training process.



(a) MNIST (b) FMNIST

(c) CIFAR10 (d) CIFAR100

Figure 2: The number of training rounds required to achieve the corresponding accuracy on different datasets.

Experiment

In this section, we comprehensively evaluate the FedAvg (Pei et al. 2024), FedADP (Liu et al. 2023), CS (Jiang and Borcea 2023), HL-SGD (Guo et al. 2022), ZeroFL (Qiu et al. 2022), and our FedCPD algorithms, covering metrics such as test accuracy, average loss, FLOPs, and communication volume. We conducted comparative experiments using four representative datasets: MNIST, Fashion MNIST, CIFAR-10, and CIFAR-100, with CNN and ResNet18 architectures. As shown in Figure 2, our FedCPD algorithm consistently outperforms other algorithms in terms of test accuracy and exhibits lower loss values across multiple datasets. Table 1 presents the test accuracy, communication volume, and FLOPs of the FedCPD algorithm at specific rounds for the four datasets. It is clear that the FedCPD algorithm maintains high test accuracy while requiring lower communication volume and FLOPs, demonstrating its effectiveness in reducing communication overhead.

Conclusion

The FedCPD algorithm reduces communication volume and computational load by utilizing pruning masks and the Hadamard product while maintaining high accuracy. Additionally, it enhances efficiency through device-to-device (D2D) communication, effectively lowering communication overhead in federated learning.

References

Guo, Y.; Sun, Y.; Hu, R.; and Gong, Y. 2022. Hybrid local SGD for federated learning with heterogeneous communications. In *International conference on learning representations*.

Jiang, X.; and Borcea, C. 2023. Complement sparsification: Low-overhead model pruning for federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 8087–8095.

Liu, H.; Shi, Y.; Su, Z.; Zhang, K.; Wang, X.; Yan, Z.; and Kong, F. 2023. FedADP: Communication-Efficient by Model Pruning for Federated Learning. In *GLOBECOM 2023-2023 IEEE Global Communications Conference*, 3093–3098. IEEE.

Pei, J.; Liu, W.; Li, J.; Wang, L.; and Liu, C. 2024. A Review of Federated Learning Methods in Heterogeneous scenarios. *IEEE Transactions on Consumer Electronics*.

Qiu, X.; Fernandez-Marques, J.; Gusmao, P. P.; Gao, Y.; Parcollet, T.; and Lane, N. D. 2022. Zerofl: Efficient on-device training for federated learning with local sparsity. *arXiv preprint arXiv:2208.02507*.