

B. Tech. CSE Sixth Semester CS350 Mini Project - II Report

On

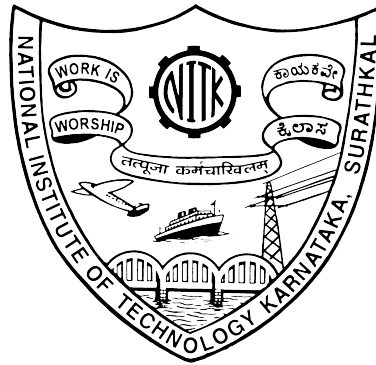
Analysis of Heterogeneous Federated Learning using WandB

Chinmay Sharma T

(211CS114)

Guide

Prof. Annappa B



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA,

SURATHKAL, MANGALORE - 575025

April, 2024

DECLARATION

I hereby declare that the B. Tech . 6th Semester **CS350 - Mini Project - II** report entitled **Analysis of Heterogeneous Federated Learning using WandB** being submitted to the Department of Computer Science and Engineering, National Institute of Technology Karnataka, Surathkal, in fulfilment of the requirements of the CS350 course is a bona fide report of the work carried out by me. The material contained in this Report has not been submitted to any University or Institution for the award of any degree.



Chinmay Sharma T

211CS114

Department of Computer Science and Engineering
NITK, Surathkal

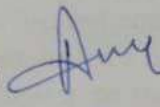
Place: NITK, Surathkal.

Date: 10-04-2024

CERTIFICATE

This is to certify that the B. Tech. 6th Semester CS350 - Mini Project - II report entitled **Analysis of Heterogeneous Federated Learning using WandB** submitted by **Chinmay Sharma T**, (Roll Number: 211CS114) as the record of the work carried out by him/her, is accepted in fulfillment of the requirements of the CS350 course.

Guide

 12/04/24

Prof. Annappa B,

Department of Computer Science and Engineering,

NITK, Surathkal

Abstract

Federated learning (FL) is an emerging paradigm to train a global deep neural network (DNN) model by collaborative clients that store their private data locally through the coordination of a central server. A major challenge is a high communication overhead during the training stage, especially when the clients are edge devices that are linked wirelessly to the central server. In this paper, we propose efficient techniques to reduce the communication overhead of FL from three perspectives.

First, to reduce the amount of data being exchanged between clients and the central server, we propose employing low-rank tensor models to represent neural networks to substantially reduce the model parameter size, leading to significant reductions in both computational complexity and communication overhead. Then, we consider two edge scenarios and propose the corresponding FL schemes over wireless channels. The first scenario is that the edge devices barely have sufficient computing and communication capabilities, and we propose a lattice-coded over-the-air computation scheme for the clients to transmit their local model parameters to the server.

Compared with the traditional repetition transmission, this scheme significantly reduces the distortion. The second scenario is that the edge devices have very limited computing and communication power, and we propose natural gradient-based FL, that involves forward pass only, and each client transmits only one scalar to the server at each training iteration. Numerical results on the MNIST data set and the CIFAR-10 data set are provided to demonstrate the effectiveness of the proposed communication-efficient FL techniques, in that they significantly reduce the communication overhead while maintaining high learning performance.

Contents

List of Figures	iv
1 Introduction	1
1.1 What is Federated Learning?	1
1.2 Citations in the Project	2
2 Federated Learning Implementation	3
2.1 Implementation using Flower and CIFAR-10 dataset	3
3 Implementation using Flower and MNIST dataset	5
4 Implementation using MNIST dataset and 6 clients	7
5 Conclusion and Future Work	10

List of Figures

1	WandB analysis of Power Usage	3
2	WandB analysis of Training Accuracy	4
3	WandB analysis of Training Accuracy	4
4	WandB analysis of Accuracy	5
5	WandB analysis of Loss	6
6	WandB analysis of Network Traffic	6
7	WandB analysis of Client-1-test loss	7
8	WandB analysis of Client-2-test loss	8
9	WandB analysis of Accuracy	8
10	WandB analysis of Test Loss	9
11	WandB analysis of System Memory Utilization	9

1 Introduction

1.1 What is Federated Learning?

Federated Learning (FL) represents a transformative paradigm in machine learning, enabling the training of models directly on decentralized data sources, such as edge devices, without the need to aggregate data centrally. This approach is particularly advantageous for resource-constrained devices that possess limited bandwidth or energy resources. By leveraging FL, these devices can contribute to model training while preserving data privacy and minimizing communication overhead.

The fundamental concept of FL involves training a global model by aggregating local updates from multiple client devices. Each client trains the model on its local data, and only model updates (not raw data) are transmitted to a central server or aggregator. This decentralized training method offers several key benefits:

1. **Privacy Preservation:** FL allows training models directly on user devices without the need to share sensitive raw data. This significantly enhances data privacy and security, a critical concern in modern machine learning applications.

2. **Communication Efficiency:** Traditional centralized machine learning requires transmitting large volumes of data to a central server, which can be impractical or costly, especially in edge environments. FL reduces communication overhead by transmitting compact model updates instead of raw data.

3. **Resource Conservation:** Edge devices often have limited computational power, memory, and battery life. FL minimizes the burden on these devices by conducting local computations and sending concise updates, thus conserving resources.

- **Decentralized Learning Algorithms:** FL algorithms can be designed to allow more autonomy and collaboration among edge devices, reducing dependency on a centralized server for model aggregation. This decentralized approach further enhances scalability and robustness in edge settings.

In summary, the evolution of FL techniques holds great promise for unlocking the potential of decentralized data sources, such as edge devices, in machine learning applications. By addressing communication efficiency challenges and emphasizing data privacy, FL paves the way for scalable and sustainable machine learning deployments in diverse and dynamic edge environments.

1.2 Citations in the Project

The development of FL techniques has garnered significant attention from both academia and industry. Notable studies include the work by Yang et al. (2019) on the concept and applications of federated machine learning [1], as well as Li et al. (2020) outlining challenges, methods, and future directions in FL [2]. Bonawitz et al. (2019) discuss system design considerations for scaling FL [3], and Federated Learning White Paper (WeBank AI Group, 2018) presents foundational principles of FL [4].

2 Federated Learning Implementation

2.1 Implementation using Flower and CIFAR-10 dataset

Our implementation involved setting up a FL system with two clients and one server using Flower, a friendly federated learning framework. Each client in our system possesses a subset of the CIFAR-10 dataset, ensuring privacy and data locality. The clients train their local models on their respective datasets and send model updates to the server periodically.

Throughout the FL process, we monitored various metrics such as model accuracy, convergence rate, and communication overhead. We observed how the global model evolves over multiple rounds of training, capturing insights into the learning dynamics and performance improvements achieved through federated learning.

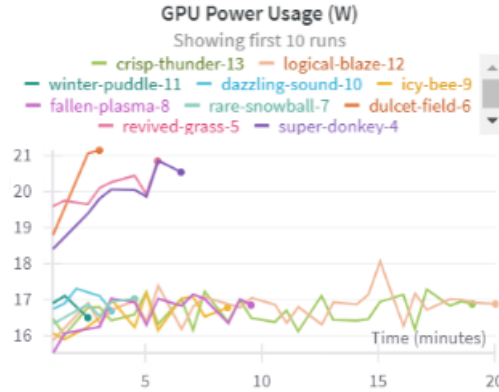


Figure 1: WandB analysis of Power Usage

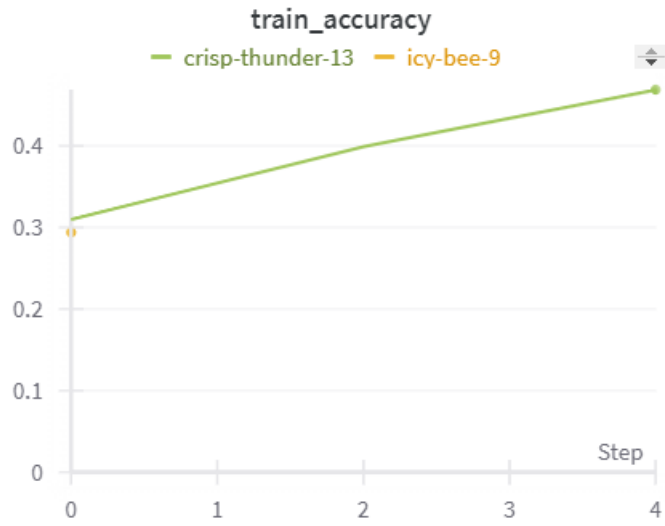


Figure 2: WandB analysis of Training Accuracy

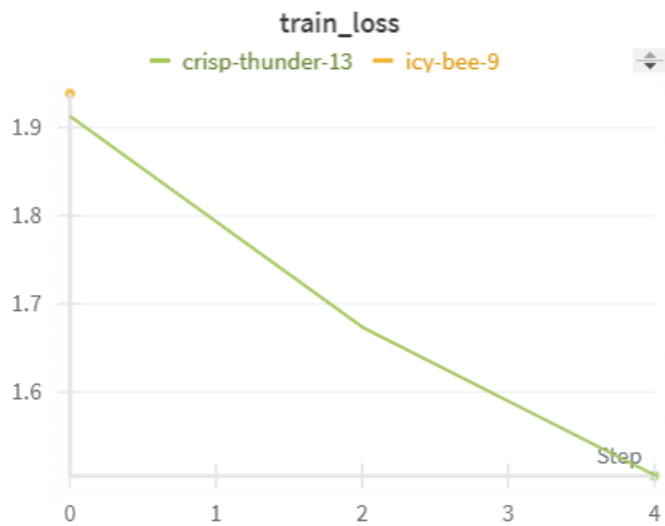


Figure 3: WandB analysis of Training Accuracy

3 Implementation using Flower and MNIST dataset

Our implementation involved setting up a federated learning (FL) system using Flower, a user-friendly federated learning framework, with two clients and one server. Each client in our system was provided with a portion of the MNIST dataset, where Client 1 was fed data representing digits 0 to 4, and Client 2 received data representing digits 5 to 9. This ensured privacy and data locality within the FL setup.

Clients independently trained their local models on their respective datasets and periodically transmitted model updates to the server. Throughout the FL process, we monitored various metrics including loss, accuracy, GPU usage, and power consumption, in addition to observing model accuracy and convergence rate. These metrics allowed us to gain insights into the learning dynamics and performance enhancements achieved through federated learning across multiple training rounds.

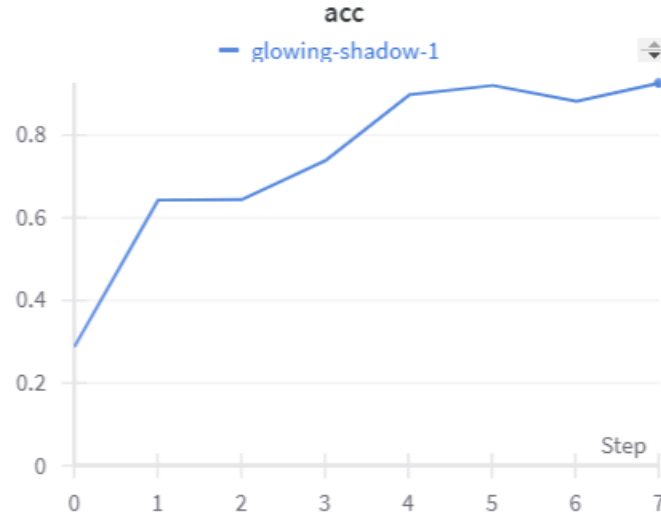


Figure 4: WandB analysis of Accuracy

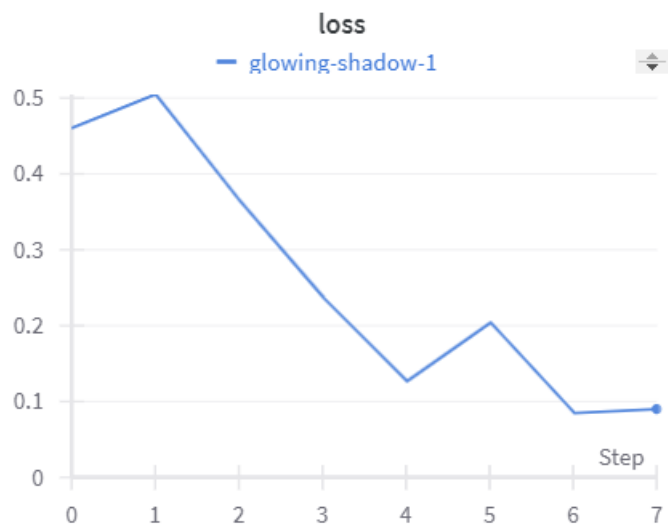


Figure 5: WandB analysis of Loss

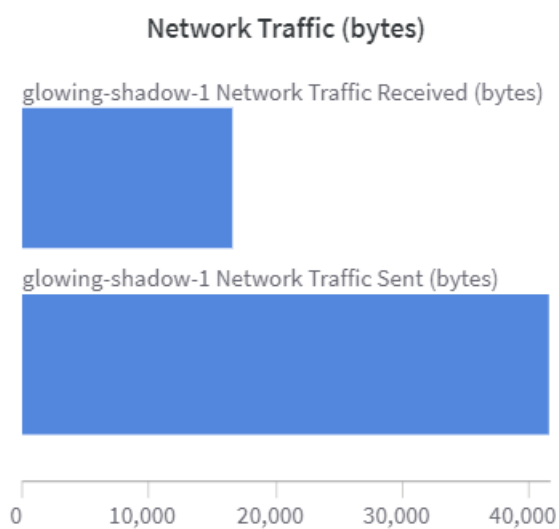


Figure 6: WandB analysis of Network Traffic

4 Implementation using MNIST dataset and 6 clients

We have implemented a federated learning (FL) system with six clients using the MNIST dataset. Each client is equipped with its own instance of the MobileNetV2 model, enabling them to train locally on their respective subsets of the dataset. This distributed approach ensures data privacy and locality while facilitating collaborative learning across the FL network.

Through integration with Wandb, we continuously monitor key metrics such as loss and accuracy during both training and evaluation phases. This real-time tracking provides valuable insights into the performance of individual clients' models, facilitating optimization and refinement of the federated learning process. The utilization of six clients enhances the scalability and efficiency of the FL system, allowing for parallelized model training and accelerated convergence.

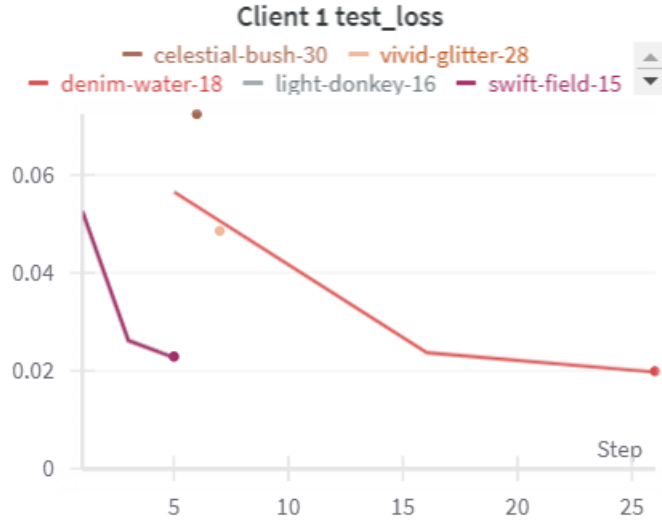


Figure 7: WandB analysis of Client-1-test loss

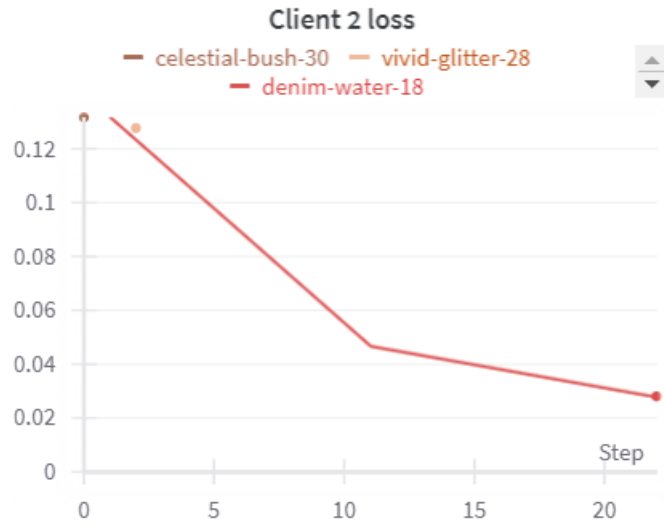


Figure 8: WandB analysis of Client-2-test loss

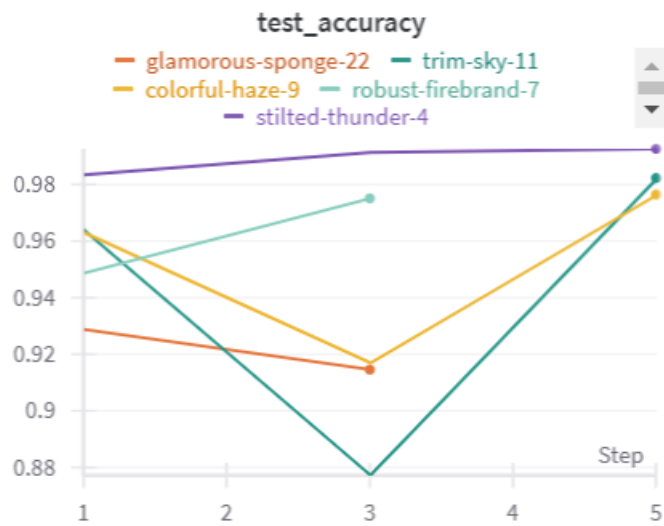


Figure 9: WandB analysis of Accuracy



Figure 10: WandB analysis of Test Loss

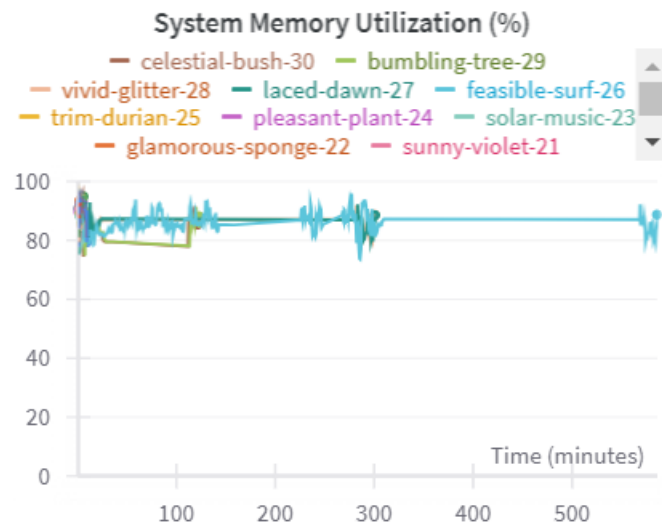


Figure 11: WandB analysis of System Memory Utilization

5 Conclusion and Future Work

In this study, we have explored the potential of Federated Learning (FL) as a transformative approach to training machine learning models on decentralized data sources, particularly in resource-constrained edge environments. Our focus has been on developing communication-efficient FL techniques that reduce overhead while maintaining model accuracy and preserving data privacy.

Through our investigation, we have highlighted the key advantages of FL, including its ability to leverage local updates from client devices, thereby minimizing the need for central data aggregation and reducing communication demands. FL holds significant promise for edge computing scenarios where devices have limited bandwidth, energy, and computational resources.

Our findings underscore the importance of privacy-preserving techniques in FL, ensuring that sensitive user data remains on local devices while contributing to model training. We have also explored advancements like natural gradient-based approaches to streamline model updates and enhance communication efficiency.

Looking ahead, there are several exciting avenues for future research and development in Federated Learning, particularly focusing on resource-constrained edge devices:

1. **Optimizing Model Compression:** Develop more efficient model compression techniques tailored for edge devices to reduce the size of model updates transmitted during FL. This will minimize communication overhead while maintaining model performance.
2. **Edge-Aware FL Algorithms:** Design FL algorithms that are specifically optimized for edge environments, considering factors like limited computational power, intermittent connectivity, and heterogeneous device capabilities.
3. **Privacy-Preserving Techniques:** Explore advanced privacy-preserving methods, such as differential privacy and secure aggregation, to further enhance data privacy in FL settings on edge devices.