Implement FedAvg from Scratch

Mohammadjavad Maheronnaghsh

UNIVERSITY OF TORONTO

Introduction

Federated Learning is a decentralized learning approach that aims to train a global model from clients with local datasets

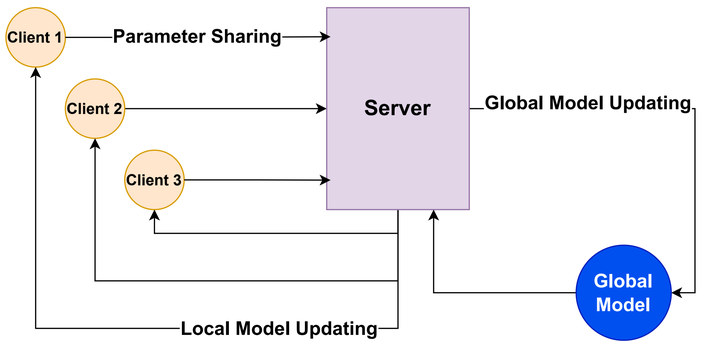

Figure 1. Overview of the federated learning approach [1]

Formulation

Here the formulation of the FedAvg [2] is presented.

Global Update:

$$\mathbf{w}_{t+1} = \sum_{k=1}^{K} \frac{n_k}{n} \mathbf{w}_{t+1}^k,$$

Local Update (SGD):

$$\mathbf{w}_{t+1}^k = \mathbf{w}_t - \eta \nabla F_k(\mathbf{w}).$$

Dataset

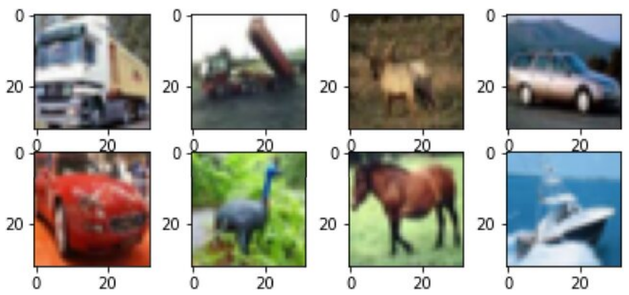

Figure 2. Examples of the CIFAR-10 dataset [3]

Model Structure

Here the structure of the simple used model is shown. It's a 2-layer CNN.

```
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Conv2d-1          [-1, 32, 32, 32]             320
         MaxPool2d-2          [-1, 32, 16, 16]               0
            Conv2d-3          [-1, 64, 16, 16]          18,496
         MaxPool2d-4            [-1, 64, 8, 8]               0
            Linear-5                 [-1, 128]         524,416
            Linear-6                  [-1, 10]           1,290
================================================================
Total params: 544,522
Trainable params: 544,522
Non-trainable params: 0
----------------------------------------------------------------
Input size (MB): 0.00
Forward/backward pass size (MB): 0.47
Params size (MB): 2.08
Estimated Total Size (MB): 2.55
----------------------------------------------------------------
```

Figure 3. Overview of the model structure

Time Analysis

Table 1. Effect of Parameters on Training Time

Parameter	Value	Training Time (s)
Batch Size	128	1881.10
Batch Size	64	2010.89
Batch Size	32	2285.47
Batch Size	4	5019.63
Client	10	2661.12
Client	100	2682.58
Client	1000	3091.82
Epoch	10	5288.18
Epoch	100	51735.66
Round	10	560.53
Round	100	5424.15
Round	1000	53381.07

Results

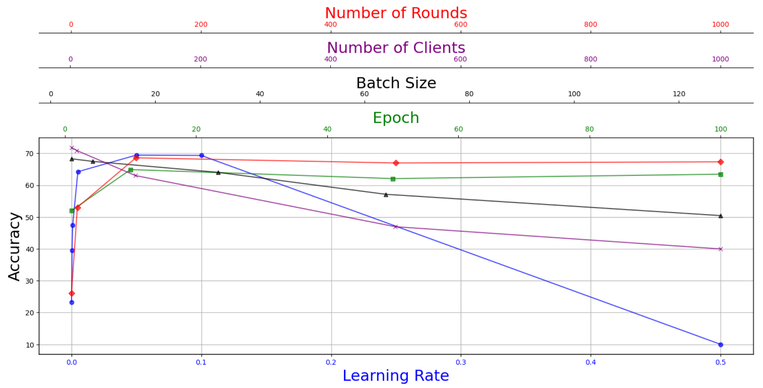

Figure 4. Overview of the results

Discussion and Conclusion

- **Learning Rate**: The learning rate must be carefully tuned as there is an optimal point.
- **Batch Size**: Larger sizes were faster but less accurate.
- **Clients**: More clients slowed training and needs more rounds and epochs.
- **Epochs**: Higher epochs improved accuracy but greatly increased time.
- **Rounds**: Fewer rounds were faster but less accurate.
- **Optimal**: Moderate settings balanced time and accuracy.

References

[1] diagrams.net (formerly draw.io).
`https://www.drawio.com/`.
Accessed: 2024-12-09.

[2] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas.
Communication-efficient learning of deep networks from decentralized data.
In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

[3] Sagnik Sarkar, Shaashwat Agrawal, Thar Baker, Praveen Kumar Reddy Maddikunta, and Thippa Reddy Gadekallu.
Catalysis of neural activation functions: Adaptive feed-forward training for big data applications.
Applied Intelligence, 52(12):13364–13383, 2022.