Recycling Model Updates in Federated Learning: Are Gradient Subspaces Low Rank?

Sheikh Shams Azam, Seyyedali Hosseinalipour, Qiang Qiu, Christopher Brinton

School of ECE, Purdue University

10th International Conference on Learning Representations (ICLR), 2022

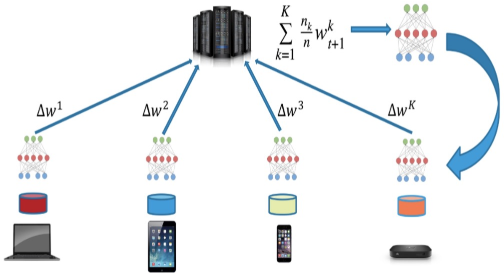
Outline

- Motivation
- Background
- Our Methodology: Look-Back Gradient Multiplier
- Theoretical and Experimental Results
- Summary

Motivation

Communication overhead in Federated Learning

- Federated learning advocates parameter sharing instead of data sharing to promote data privacy.
- Deep models have millions/billions of parameters and impose communication burden for federated systems.
- Increasing interest in reducing model size or communication frequency.



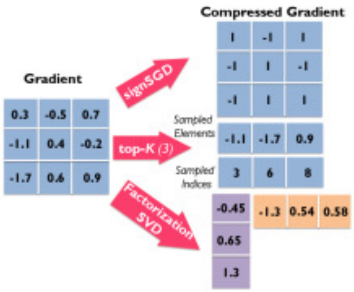
Source: https://medium.com/nerd-for-tech/build-your-own-federated-learning-model-2c882ea8cfde

Devices share the model parameters/gradients with the server which are then averaged and synchronized across devices before next local update.

Background

Gradient Compression

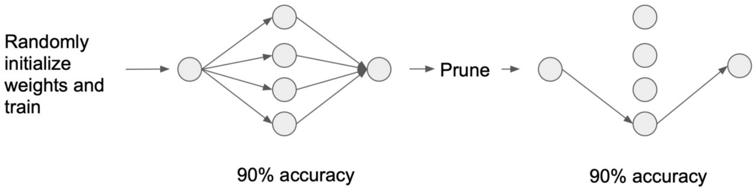
1. Sparsification of gradient components with small magnitude, e.g., top-k sparsification.
2. Quantization of magnitude of gradient components, e.g., SignSGD [1], Q-SGD [2].
3. Low-rank approximations (e.g., using SVD) of gradient matrices, e.g, ATOMO [3], PowerSGD [4].



Source: https://deepai.org/publication/on-the-utility-of-gradient-compression-in-distributed-training-systems

Overparameterization of Neural Networks

1. Analysis of Hessian during SGD argues that SGD happens in a small subspace (i.e., with few basis vectors) [5, 6].
2. Several papers [7, 8] advocate that a majority of neurons in the neural network are insignificant to the performance.



Source: https://towardsdatascience.com/breaking-down-the-lottery-ticket-hypothesis-ca1c053b3e58

[1] Siede et al. 1-bit SGD and its application to data-parallel distributed training of speech dnns. INTERSPEECH, 2014.
[2] Alistarh et al. QSGD: Communication-efficient SGD via gradient quantization and encoding. NeurIPS, 2017.
[3] Wang et al. ATOMO: Communication-efficient Learning via Atomic Sparsification. NeurIPS, 2018.
[4] Vogels et al. PowerSGD: Practical low-rank gradient compression for Distributed Optimization. NeurIPS, 2019.
[5] Sagun et al. Eigenvalues of the Hessian in Deep Learning: Singularity and Beyond. ArXiv, 2016.
[6] Ghorbani et al. An investigation into Neural Net Optimization via Hessian Eigenvalue Density. ICML, 2019.
[7] Frankle et al. The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks. ICLR, 2019
[8] Liu et al. Rethinking the Value of Network Pruning. ICML, 2019.

Our Methodology: Look-Back Gradient Multiplier

Contributions of LBGM

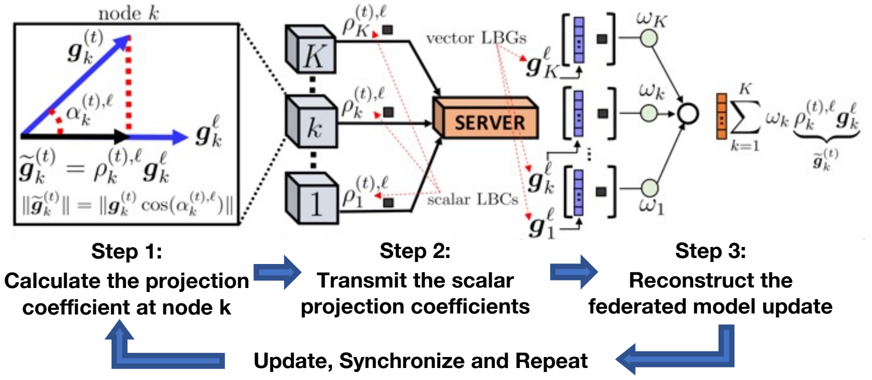
1. Study the principal components of the gradient subspace and propose the hypotheses:

   H1: The subspace spanned by gradients generated across SGD are low-rank.

   H2: The Principal Gradient Directions (PGDs) can be approximated using a subset of gradients generated across SGD epochs.

2. Using H1 & H2, we develop LBGM that significantly reduces communication overhead in FL.
3. Characterize the theoretical convergence of LBGM and demonstrate its experimental benefits.

'Under LBGM nodes transmit **scalar look-back coefficients (LBCs)** instead of **millions/billions of model parameters/gradients.**'



Look-Back Gradient Multiplier (LBGM)

Theoretical and Experimental Results

Rank Characteristics of SGD

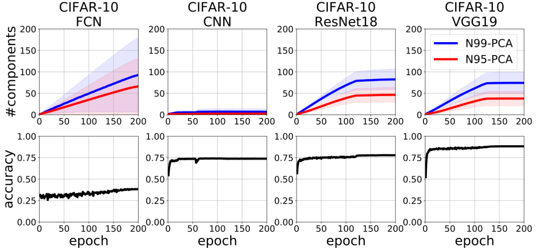


Figure 1: *PCA Component Progression on CIFAR-10*: The top row shows number of components that account for 99% (blue) and 95% (red) explained variance of all the gradients generated during SGD epochs. The bottom row shows the performance of the model on test (held-out) data.

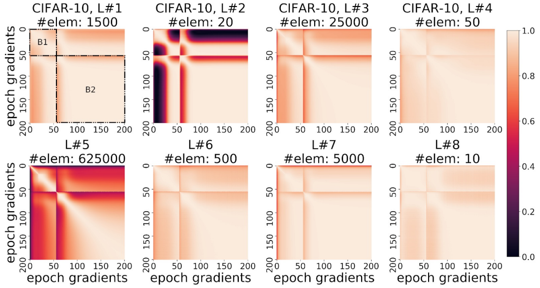


Figure 2: *Similarity among consecutive gradients*: The newly generated gradients can be represented in terms of the previously generated gradients with low approximation error, thus suggesting that gradients can be recycled in federated learning.

Theoretical Results

Theorem 1 (refer paper) characterizes the behavior of LBGM: Performance of LBGM increases as error threshold (a tunable parameter) is decreased.

Corollary 1 (refer paper) guarantees the convergence of the LBGM algorithm unless the magnitude of error exceeds the norm of the gradient itself.

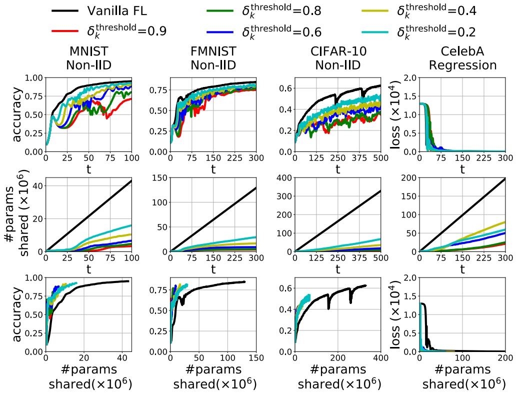


Figure 3: Effect of Error Threshold on LBGM: For larger values of error threshold, LBGM achieves communication benefits (middle row) while maintaining performance identical to vanilla FL (top row).

Experimental Results

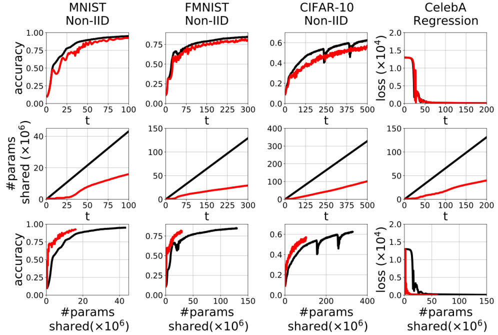


Figure 4 (Standalone): LBGM (red) consistently outperforms vanilla FL (black) in terms of number of parameters shared (middle row).

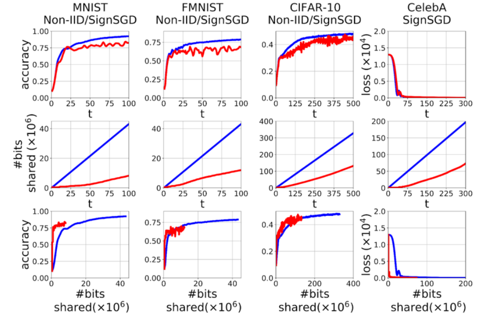


Figure 5 (Plug-n-play): SignSGD w/ LBGM (red) consistently outperforms SignSGD w/o LBGM (black) in terms of number of parameters shared (middle row).

Summary

- Federated learning and communication overhead
- Gradient compression and overparameterizations of neural networks
- Our methodology: LBGM and its novel contributions
- Theoretical and experimental results

Thank you!

Questions?

Contact: azam1@purdue.edu

Or refer to our paper:

https://openreview.net/pdf?id=B7ZbqNLDn-_