A Survey on Federated Learning

Adira Cohen, Qiurui Du, Miles Woollacott

North Carolina State University

Spring 2025

Table of Contents

Why Federated Learning?

Suppose we want to detect brain tumors from various MRI scans of brains, where the data originates from several hospitals. This is a binary classification problem, so we could use Logistic Regression.

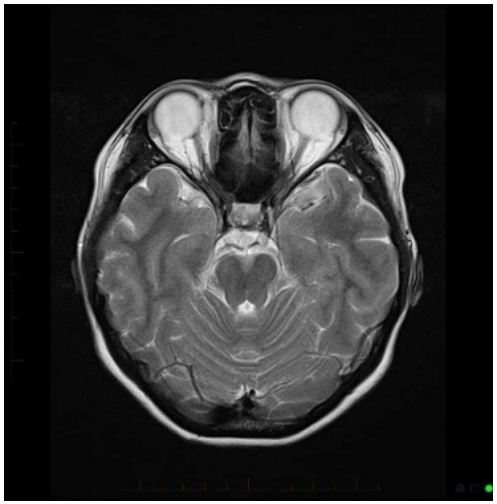

Figure: MRI of a brain (Melbourne Radiology Clinic)

Why Federated Learning?

Suppose we want to detect brain tumors from various MRI scans of brains, where the data originates from several hospitals. This is a binary classification problem, so we could use Logistic Regression.

So, what are the problems?

- Data privacy is an issue.
  - Federal laws prevent aggregation of medical imaging data [Guan et al., 2024].
  - Data must remain decentralized.

Why Federated Learning?

Suppose we want to detect brain tumors from various MRI scans of brains, where the data originates from several hospitals. This is a binary classification problem, so we could use Logistic Regression.

So, what are the problems?

- Data privacy is an issue.
  - Federal laws prevent aggregation of medical imaging data [Guan et al., 2024].
  - Data must remain decentralized.
- Data is not i.i.d, and datasets can have different sizes.
  - Medical image datasets collected from specific hospitals reflect the local population. [Guan et al., 2024]
  - An assumption of distributed learning is i.i.d. data!

Main Idea of Federated Learning (FL)

Parties collaborate with each other to train an ML model without explicitly sharing data.

- **Parties/Clients/Devices**: Entities that host a local database.
- Back and forth communication between central server (aggregator) and parties with *updates* to the model.

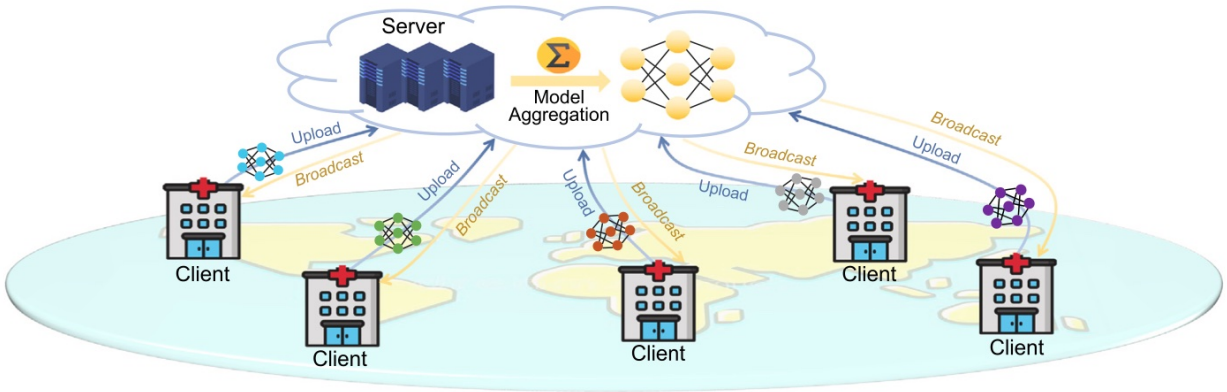

Figure: FL applied to medical imaging [Guan et al., 2024]

FedAvg

- McMahan et al. [2017]: Seminal paper on FL.
- Describes how to train neural networks on decentralized data while minimizing communication costs via the FedAvg algorithm.
- **FedAvg**: Each party does one or more rounds of SGD, then a central server aggregates the results.

FedAvg

Algorithm 1 `FederatedAveraging`. The K clients are indexed by k; B is the local minibatch size, E is the number of local epochs, and η is the learning rate.

Server executes:
initialize w_0
for each round $t = 1, 2, \dots$ **do**
  $m \leftarrow \max(C \cdot K, 1)$
  $S_t \leftarrow$ (random set of m clients)
  for each client $k \in S_t$ **in parallel do**
    $w_{t+1}^k \leftarrow \text{ClientUpdate}(k, w_t)$
  $w_{t+1} \leftarrow \sum_{k=1}^{K} \frac{n_k}{n} w_{t+1}^k$

ClientUpdate(k, w): *// Run on client k*
$\mathcal{B} \leftarrow$ (split $\mathcal{P}_k$ into batches of size B)
for each local epoch i from 1 to E **do**
  for batch $b \in \mathcal{B}$ **do**
    $w \leftarrow w - \eta \nabla \ell(w; b)$
return w to server

Figure: FedAvg algorithm [McMahan et al., 2017].

FL to Motivating Example

Algorithm FL for Brain Tumor Image Classification

1: **while** not converged **do**
2: Obtain list of hospitals that can participate in this iteration.
3: Central server sends $\hat{\beta}$ to hospitals. Each hospital trains their model with $\hat{\beta}$.
4: Hospitals calculate gradient vectors of $\hat{\beta}$ that correspond to a direction to move in, sends gradient to central server.
5: Central server aggregates gradient vectors in some way.
6: **end while**

Advantages and Disadvantages of FL

Advantages:

- Maintains data privacy.
  - Data is never shared between parties and the central server.
- Collaborative learning without data aggregation.
- Can personalize global model to each party [Ludwig and Baracaldo, 2022].

Advantages and Disadvantages of FL

Advantages:

- Maintains data privacy.
  - Data is never shared between parties and the central server.
- Collaborative learning without data aggregation.
- Can personalize global model to each party [Ludwig and Baracaldo, 2022].

Disadvantages:

- Computation time.
  - Especially a problem with neural networks [Jin et al., 2023].
- Not *totally* free from attacks/threats.

Low Parameter - FL (Jiang et al. [2024])

Motivation: Large-language models (LLMs) are useful for improving phone usability. However, LLMs are too expensive to train with standard FL methods using phones because of the (extremely) large number of parameters. Also, most of the phone data is unlabeled.

Idea: Do we actually need to train all of the parameters at once?

Low Parameter - FL (Jiang et al. [2024])

Algorithm

- **Global model update:**
  1. Ask each client to update
  2. Take the federated average
  3. Send the update back to the clients
- **Client update:**
  1. Example of semi-supervised learning
  2. Label some of the data
  3. Train using Low-Rank Adaptation (LoRA) (Hu et al. [2021]) for parameter fine-tuning
     - Freezes parameters to greatly reduce the number of trainable parameters

Low Parameter - FL (Jiang et al. [2024])

Results:

- Can greatly reduce the number of parameters trained, less than 0.25%
- Decreases potential learning but avoids overfitting
- Performs very well on classic test datasets, only slightly worse than centralized training
- Can outperform the centralized version and full-parameter versions
  - Centralized version is training on a larger number of incorrect labelings at once compared to local training, reducing overfitting to incorrect labelings
  - Full parameter version can remember noise, i.e. train on incorrect labelings

FedFed Algorithm (Yang et al. [2023])

Motivation: Often the data includes some features which are critical to model performance or very heterogeneous across sites while the remaining features have less of an impact or are common between sites.

Idea: What if we could share these important features globally while preserving privacy?

FedFed Algorithm (Yang et al. [2023])

The FedFed Algorithm at a glance:

- Split data into **performance-sensitive** and **performance-robust** features
- **Add noise** to the **performance-sensitive** features to preserve privacy and then **share globally**

FedFed Algorithm (Yang et al. [2023])

Splitting features: Given valid partition $X = X_r + X_s$ and labels Y, we say X_s is a performance-sensitive feature and X_r is a performance-robust feature if

$$I(X; Y|X_s) = 0$$

for information entropy H and mutual information I

Therefore to find minimal sufficient information we want to solve

$$X_s = min_Z I(X; Y|Z) \text{ s.t. } I(X; X - Z|Z) \geq I_{FF}$$

for some constant I_{FF}

FedFed Algorithm (Yang et al. [2023])

Results: (In certain situations) FedFed can significantly increase the accuracy (one over 40%!) and convergence rate

FedFed didn't improve the model when

- FL already performs similarly to centralized learning
- The noise necessary to preserve privacy was very large

Poisoning Attack Protection (Yazdinejad et al. [2024])

Motivation:

- There are malicious entities trying to poison the model
- But the attacks are difficult to detect because the gradients are not IID and are encrypted
- How do we protect against poison attacks and protect privacy without sacrificing computation/communication time or accuracy?

Idea:

- Combine multiple poisonous gradient detection algorithms
- Add redundancy to auditing system
- Use stochastic gradient descent with momentum
- Remove redundant gradients before aggregating

Poisoning Attack Protection (Yazdinejad et al. [2024])

Malicious gradient detection:

- **Gaussian Mixture Models:** Good at identifying clusters of data in heterogeneous datasets
- **Mahalanobis Distance:** Measures the distance between a point and a distribution, scale-invariant, accounts for covariance within data; used to detect unusual gradients

Poisoning Attack Protection (Yazdinejad et al. [2024])

Auditing redundancy:

- Auditors are the main trusted security entity which holds the encryption keys
- All gradients are passed through the auditor before being aggregated on the central server
- Redundancy added in one of two ways
  1. **Independent auditing:** Each auditing entity acts independently
  2. **Consensus auditing:** The auditing entities must come to a consensus about the gradient

Poisoning Attack Protection (Yazdinejad et al. [2024])

Optimization

- **Stochastic gradient descent with momentum:**
  - Improves security while also improving training efficiency
  - Applied during local training
  - Takes previous gradients' trajectories into account
  - Improves convergence rate and reduces variance
- **Pre-aggregation size reduction:**
  - Remove redundant gradients before aggregating
  - Schedule communication from data owners to allow more training steps between communications
- **Encryption:** Computes least-computationally expensive encryption necessary to maintain privacy

Poisoning Attack Protection (Yazdinejad et al. [2024])

Results: The proposed model greatly outperforms competitors at identifying targeted and untargeted attacks with a 50% attack rate

Sample frame title

Federated learning is mad useful.

References I

H. Guan, P.-T. Yap, A. Bozoki, and M. Liu. Federated learning for medical image analysis: A survey. *Pattern Recognition*, page 110424, 2024.

E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. Lora: Low-rank adaptation of large language models, 2021. URL `https://arxiv.org/abs/2106.09685`.

J. Jiang, H. Jiang, Y. Ma, X. Liu, and C. Fan. Low-parameter federated learning with large language models. In *International Conference on Web Information Systems and Applications*, pages 319–330. Springer, 2024.

References II

Y. Jin, H. Zhu, J. Xu, and Y. Chen. *Federated Learning*. Springer, 2023.

H. Ludwig and N. Baracaldo. *Federated learning: A comprehensive overview of methods and applications*. Springer, 2022.

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*, pages 1273–1282. PMLR, 2017.

Melbourne Radiology Clinic. Mri scan – brain. https://www.melbourneradiology.com.au/diagnostic-imaging/mri-scan-brain/, 2022.

References III

Z. Yang, Y. Zhang, Y. Zheng, X. Tian, H. Peng, T. Liu, and B. Han. Fedfed: Feature distillation against data heterogeneity in federated learning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 60397–60428. Curran Associates, Inc., 2023. URL `https://proceedings.neurips.cc/paper_files/paper/2023/file/bdcdf38389d7fcefc73c4c3720217155-Paper-Conference.pdf`.

References IV

A. Yazdinejad, A. Dehghantanha, H. Karimipour, G. Srivastava, and R. M. Parizi. A robust privacy-preserving federated learning model against model poisoning attacks. *IEEE Transactions on Information Forensics and Security*, 19:6693–6708, 2024. doi: 10.1109/TIFS.2024.3420126.