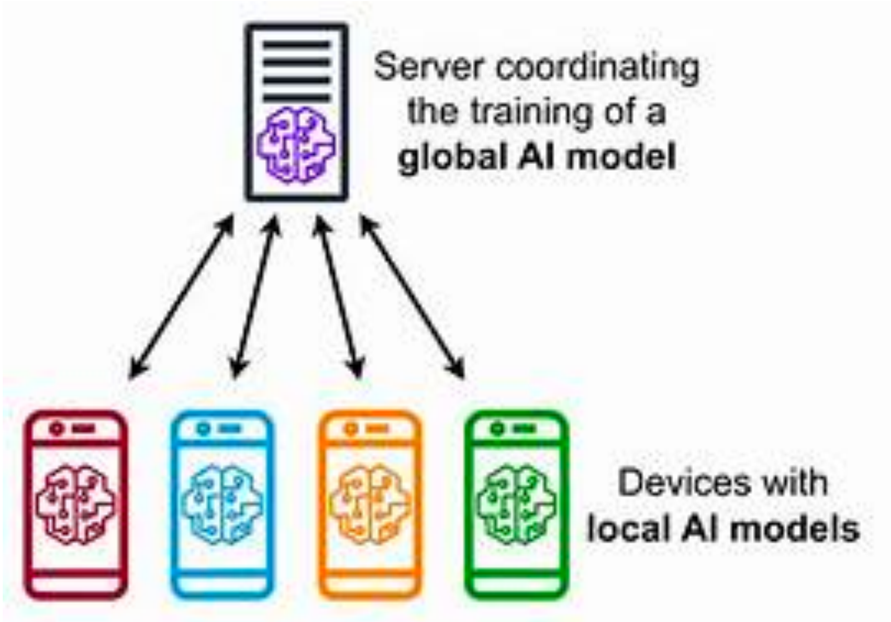
Federated Learning for Privacy-Preserving Machine Learning

Project Proposal

Raj Kumar Maurya, 2024.03.05

Introduction

- Federated Learning (FL) is a decentralized machine learning approach where model training is performed across multiple edge devices or servers holding local data, without exchanging them. This project proposal outlines the implementation of Federated Learning in a real-world scenario.



Federated Learning Algorithms

- **Federated Averaging (FedAvg):** It is one of the foundational federated learning algorithms. In FedAvg, each participating device trains its local model on its own data, and then the model updates are aggregated (averaged) at a central server. This process helps in preserving data privacy since raw data doesn't leave the devices.
- **Federated Stochastic Gradient Descent (FedSGD):** Similar to traditional SGD, FedSGD optimizes a global model by updating parameters using gradients computed locally on each device. These local gradients are then aggregated to update the global model.
- There are many other algorithms:
  - Federated Momentum (FedMo)
  - Federated Proximal (FedProx)
  - Federated Learning with Adaptive Gradient Clipping (FLAG)
  - Federated Learning with Differential Privacy (FLDP)
  - Federated Learning with Differential Privacy (FedDP)
  - Federated Learning with Secure Aggregation (FedSecAgg)
  - Federated Learning with Homomorphic Encryption (FedHomEnc)

Objective

- Understand Federated Learning: Study the principles, challenges, and benefits of federated learning.
- Implement Federated Learning techniques to leverage distributed data for model training.
- Demonstrate the feasibility of Federated Learning in improving model performance while maintaining data privacy.
- Apply federated learning algorithms to a specific use case and do comparative analysis

Methodology

- **Literature Review:** Survey existing research on federated learning
- **System Architecture:** Design the system to implement the federated learning in a simulated environment or actually distributed systems
- **Algorithm Selection:** Choose appropriate federated learning algorithms based on use case requirements.
- **Implementation:** Develop a proof-of-concept system. Dataset and the problem statement to be finalised.
- **Evaluation Metrics:** Define metrics to assess model performance, communication efficiency, and privacy preservation.

FedAvg vs FedSGD

Aspect	FedSGD	FedAvg
Aggregation Method	Aggregates gradients	Aggregates model parameters
Update Strategy	Updates global model directly with gradients	Averages model parameters to update global model
Communication Overhead	Typically lower due to gradient transmission	May have slightly higher overhead due to parameter transmission
Privacy Preservation	Gradients may contain more information about local data	Averaged parameters may preserve privacy better

Expected Outcomes

- **Functional Prototype:** A working federated learning system demonstrating privacy-preserving model training.
- **Performance Evaluation:** Comparative analysis of the two federated learning algorithms chosen.
- **Documentation:** Detailed project report, code snippets, and experimental results.

Feedbacks

- *Focus on the distributed aspects*
- *Try with smaller models*
- *Use cluster of VMs provided instead of simulating locally*
  - *Since VM was not accessible, went ahead with simulation ono multi-core system after getting confirmation from the TA (Manaswi Ma’am)*