HACCS: Heterogeneity-Aware Clustered Client Selection for Accelerated Federated Learning

Joel Wolfrath, Nikhil Sreekumar, Dhruv Kumar, Yuanli Wang, and Abhishek Chandra

Distributed Computing Systems Group

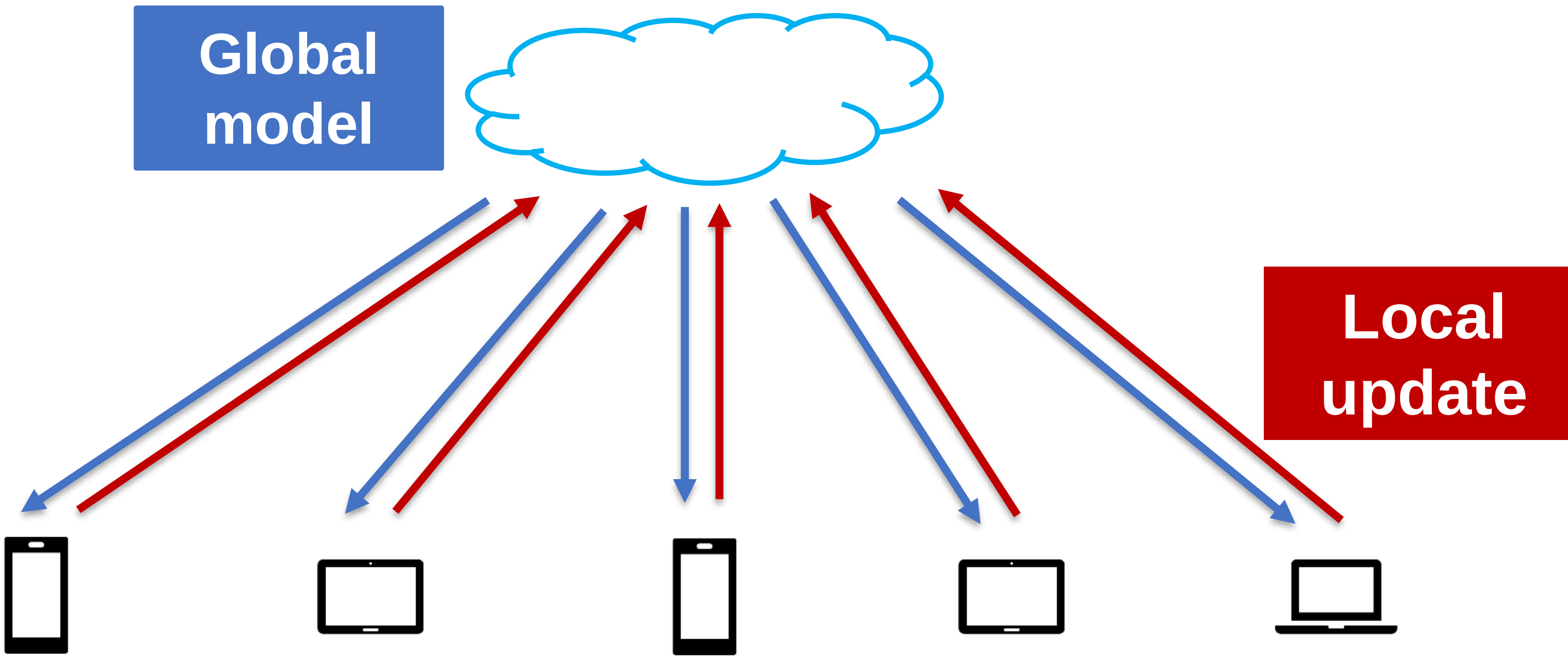
University of Minnesota
Driven to Discover[SM]

Motivation

- Data is increasingly generated in a distributed manner
- ML Applications on mobile phones
  - Next word prediction
  - Image classification

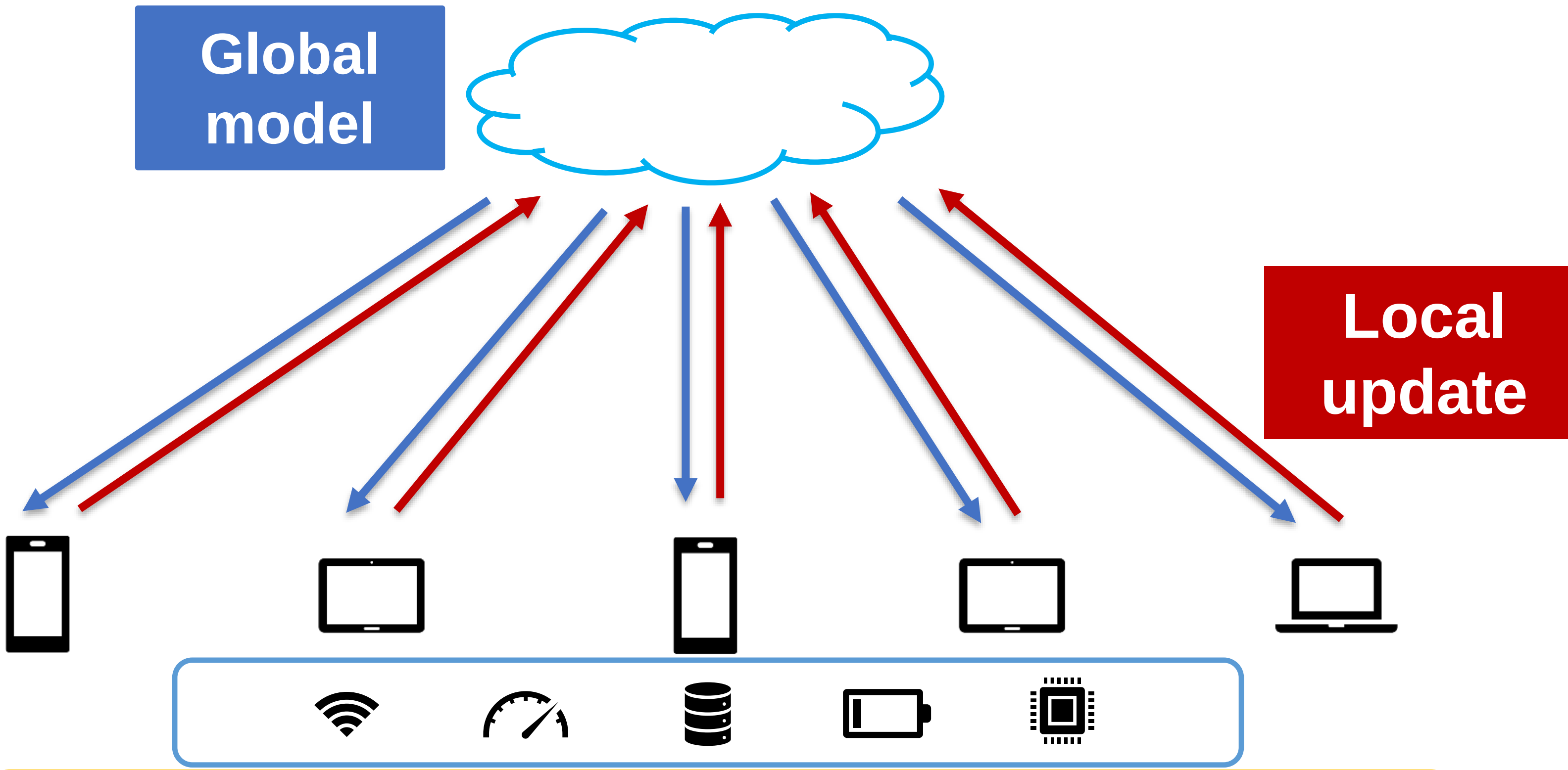
Problem: Transferring data to a central location is expensive and has privacy implications

Federated Learning



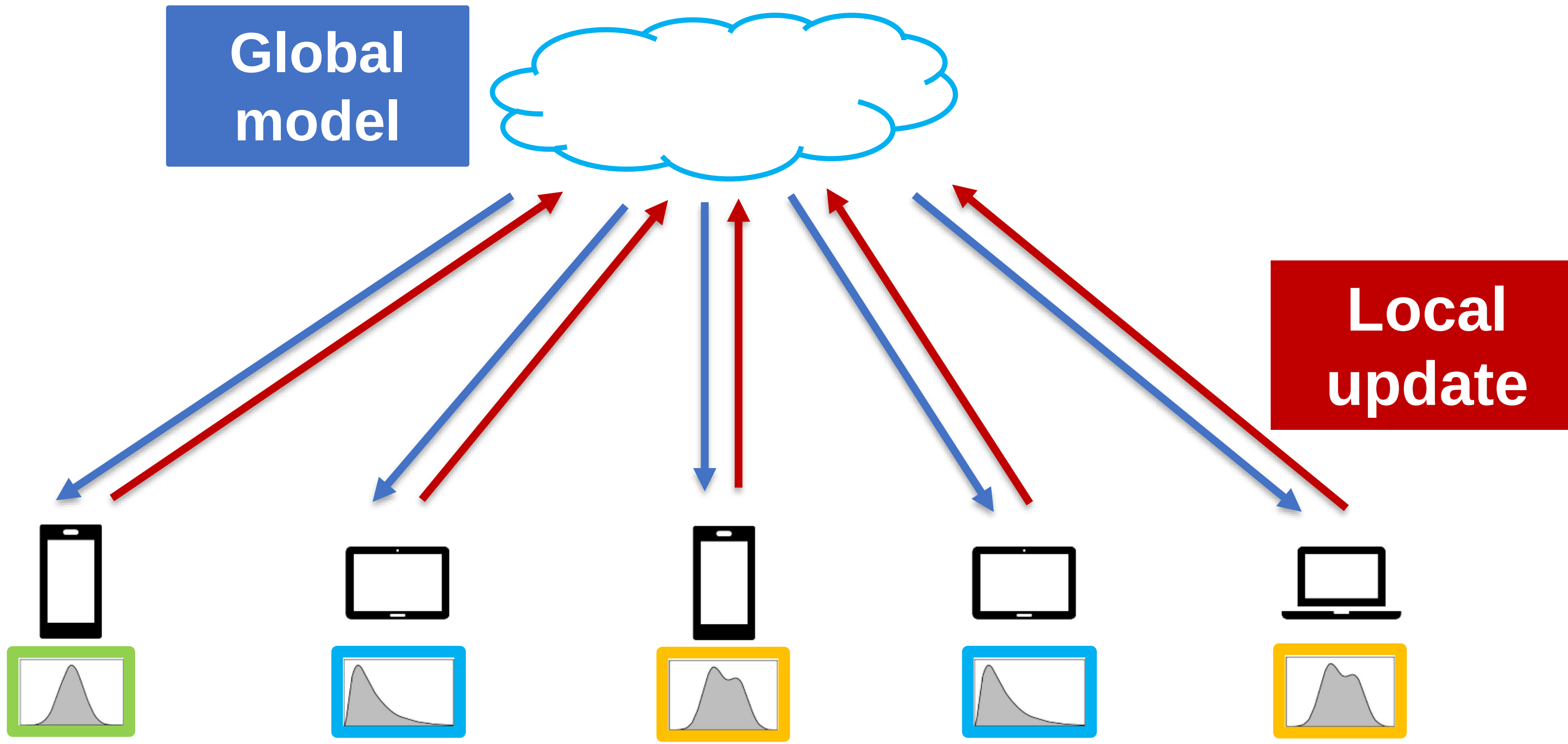
Learn a shared ML model together without uploading private training data

Federated Learning



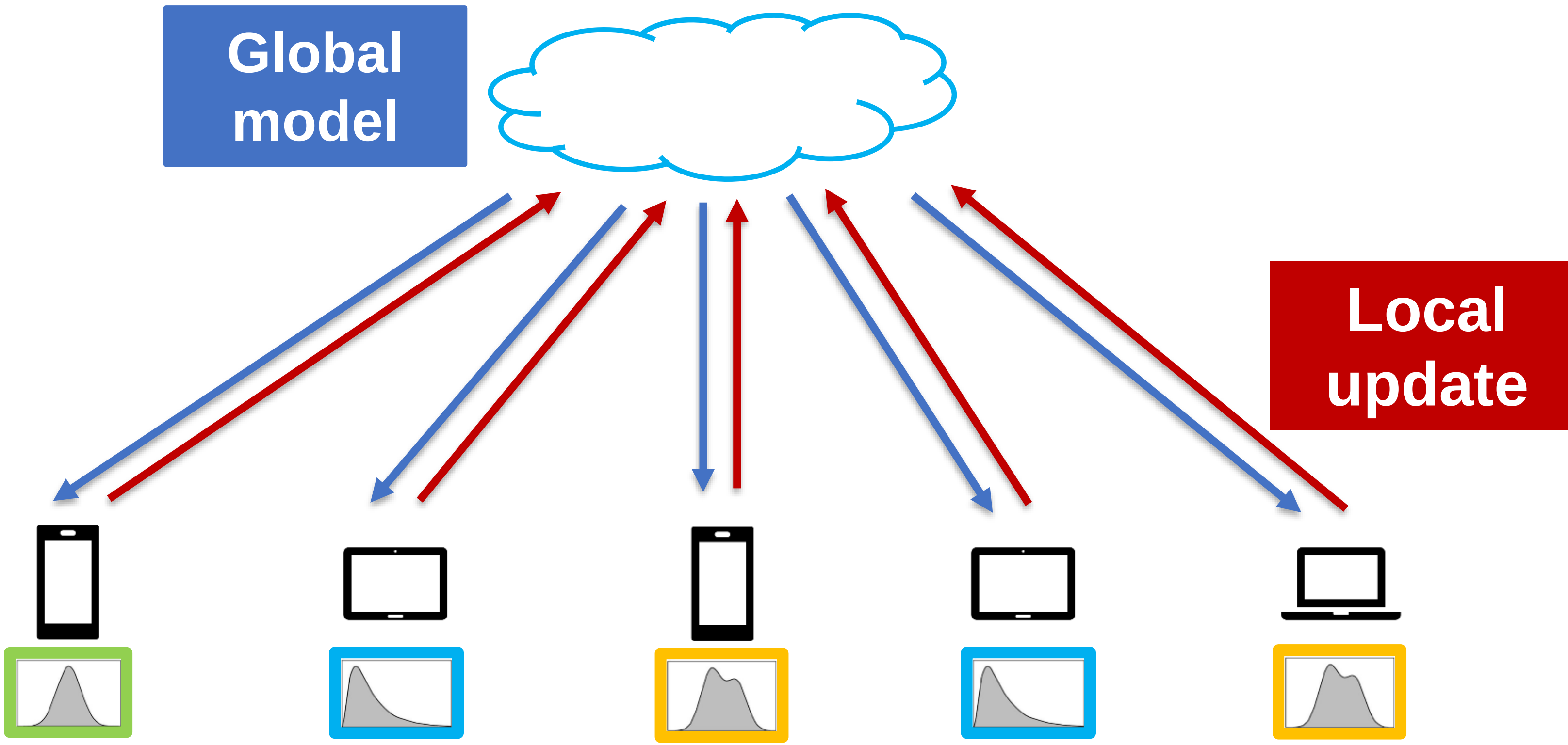
System Heterogeneity: Different devices have different computation resources

Federated Learning



Data Heterogeneity: The dataset of different devices have different statistical distributions (non-IID)

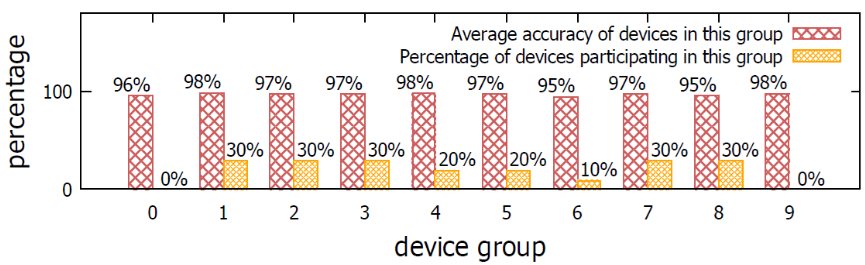
Impact of data heterogeneity



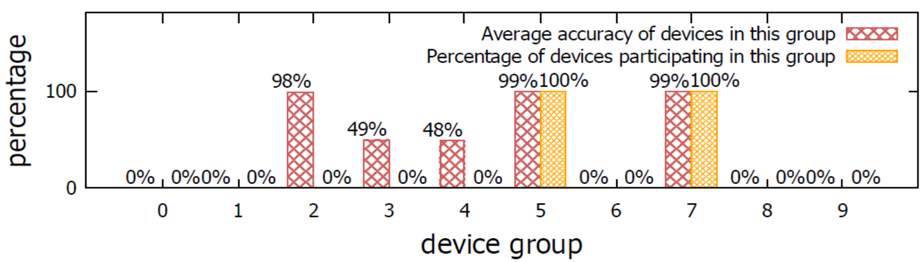
Question: What is the impact of non-IID data on Federated Learning?

Distribution Representation

- Partition 100 clients into 10 groups. Each group contains ten clients and will be assigned only two classes from MNIST dataset.
- Drop 80 out of 100 devices under 2 different patterns.
- Measure the trained global model's accuracy on the local test dataset of each device.



randomly pre-select some clients to drop

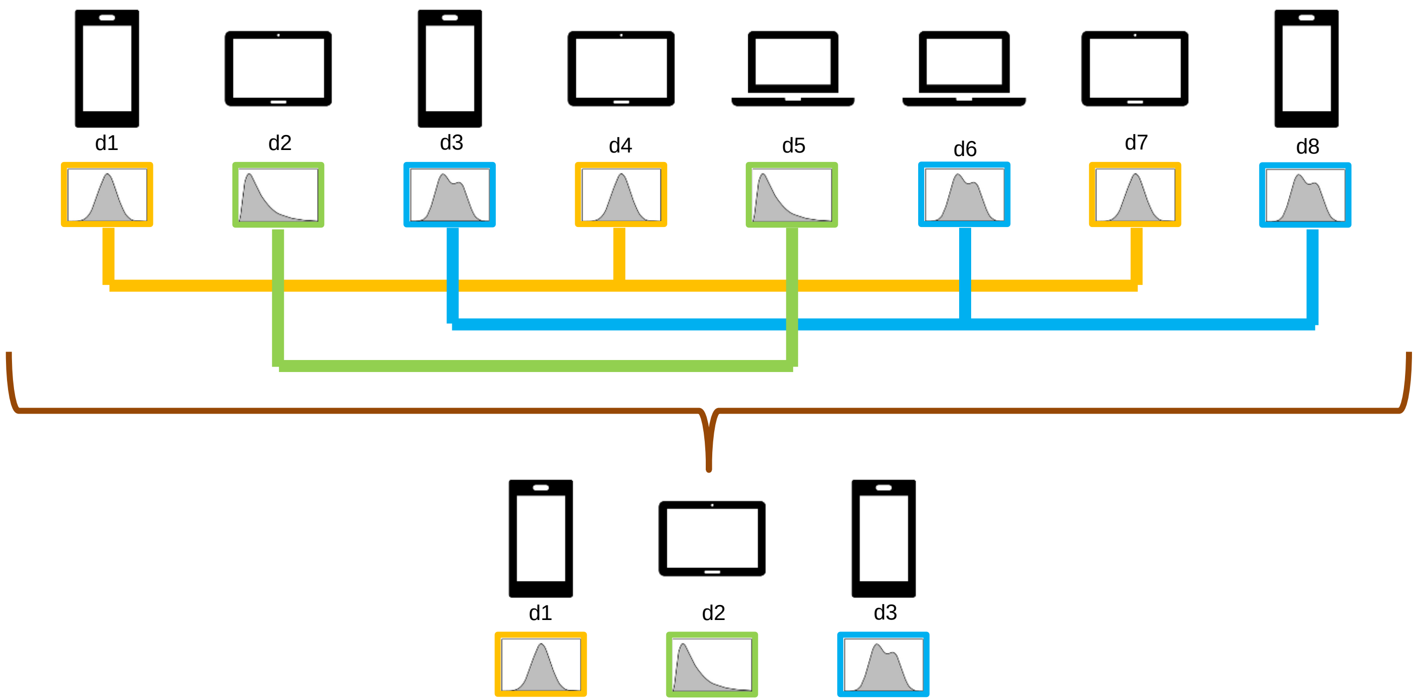


pre-select an entire group of devices to drop

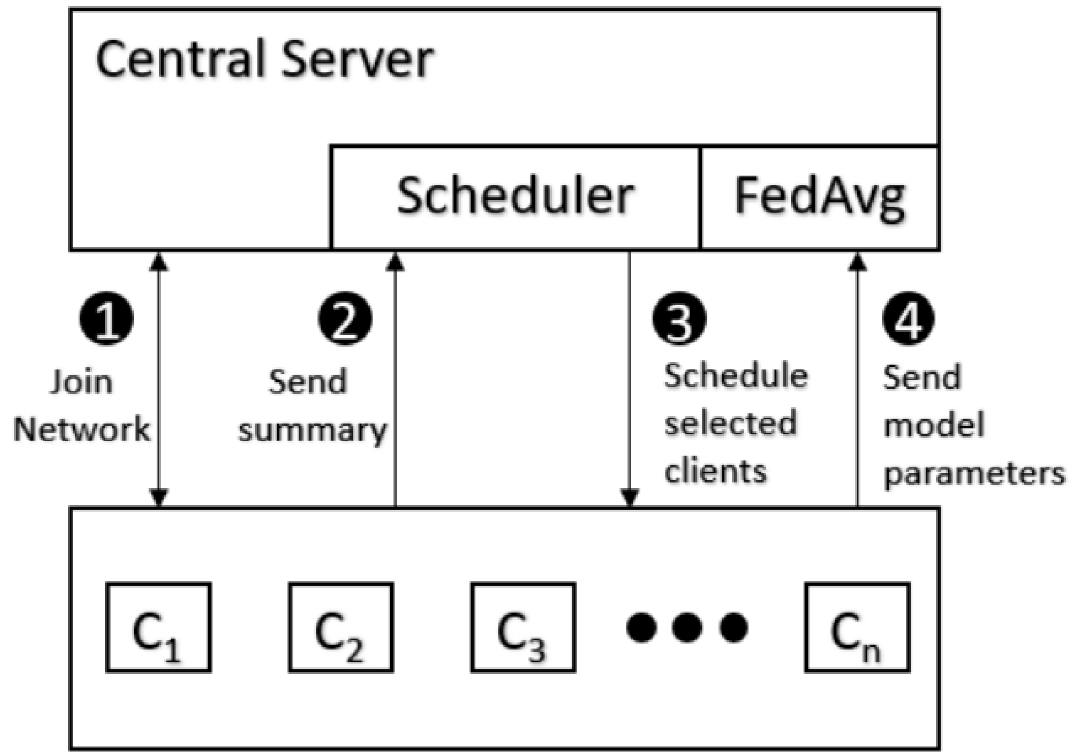
FL is robust to permanent failures, provided the data heterogeneity is well represented

Exploiting data heterogeneity

Idea: Accelerate training by identifying subsets of devices with "sufficiently similar" data distributions



System Design



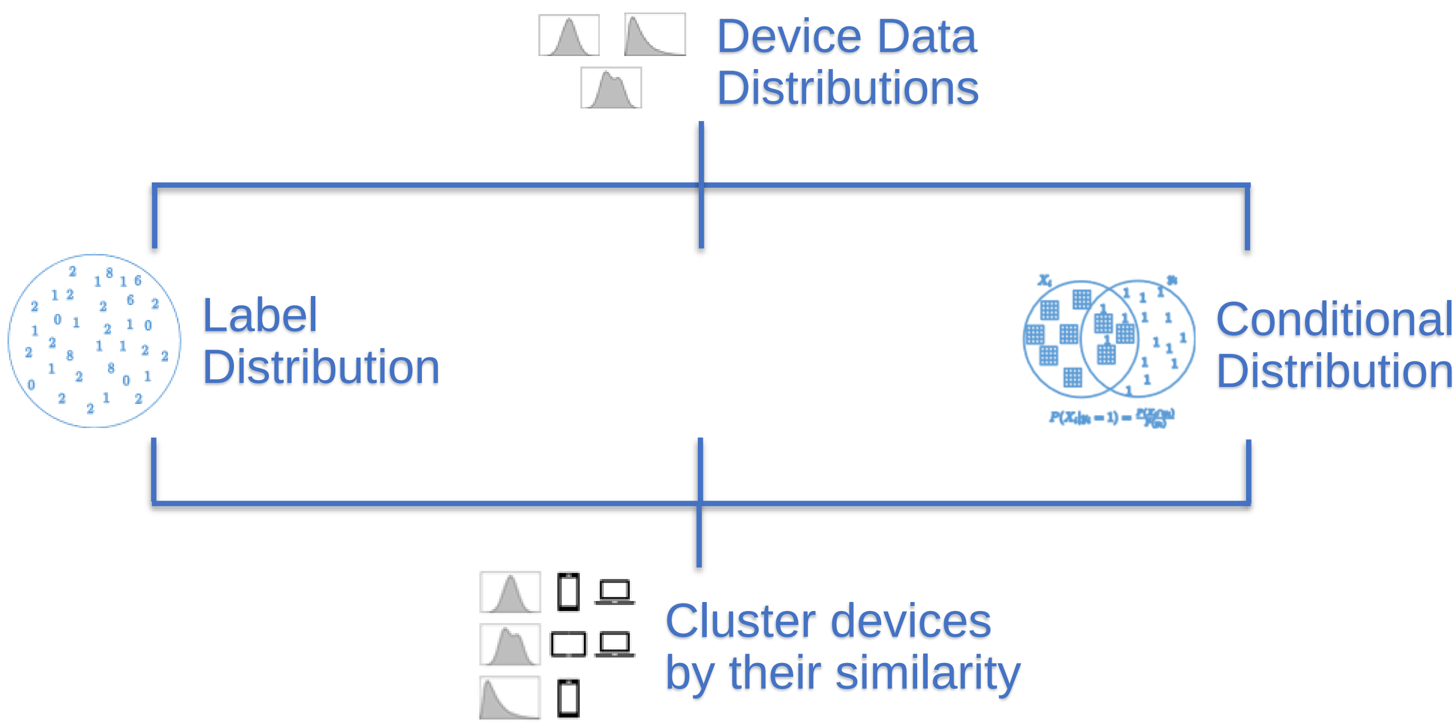
Types of IID Violations

Training data at each device drawn from a joint distribution $p(x, y)$

$$p(x, y) = p(x\,|y)\,p(y)$$

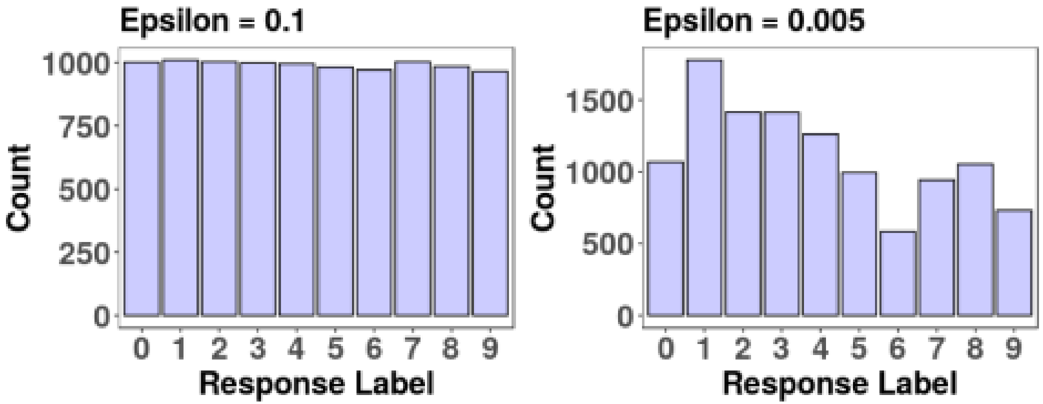
$p(y)$	Labels have different distributions
$p(x\,|y)$	Different data generates the same labels

Our Solution: Identify Data Similarity

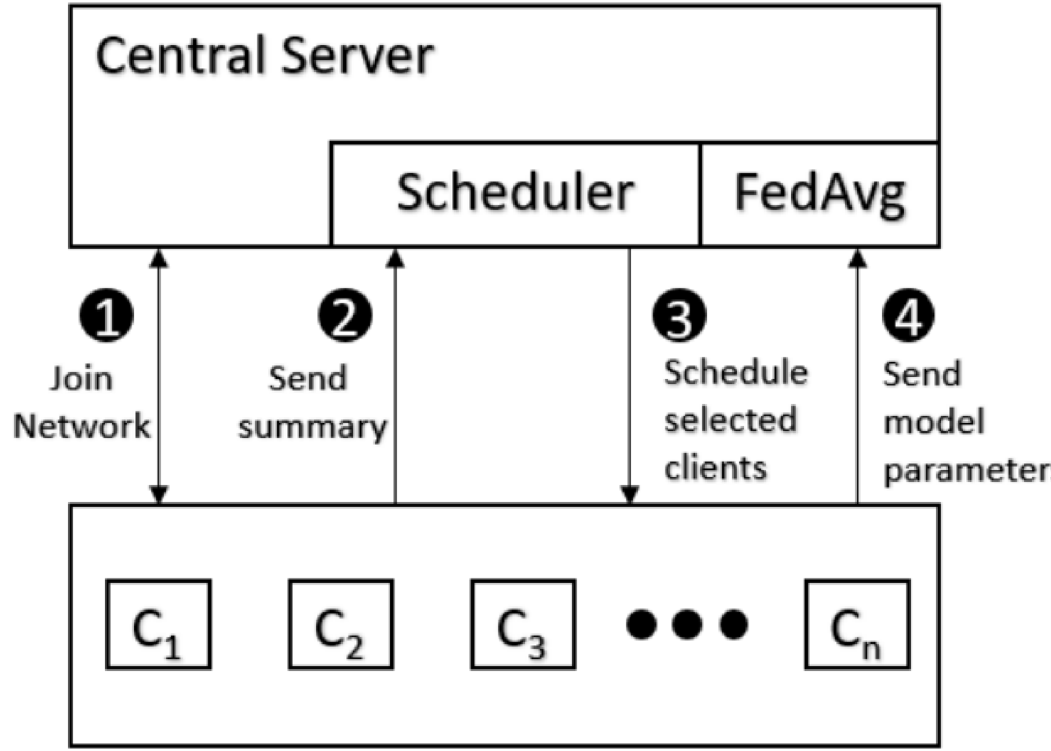


Preserving Privacy

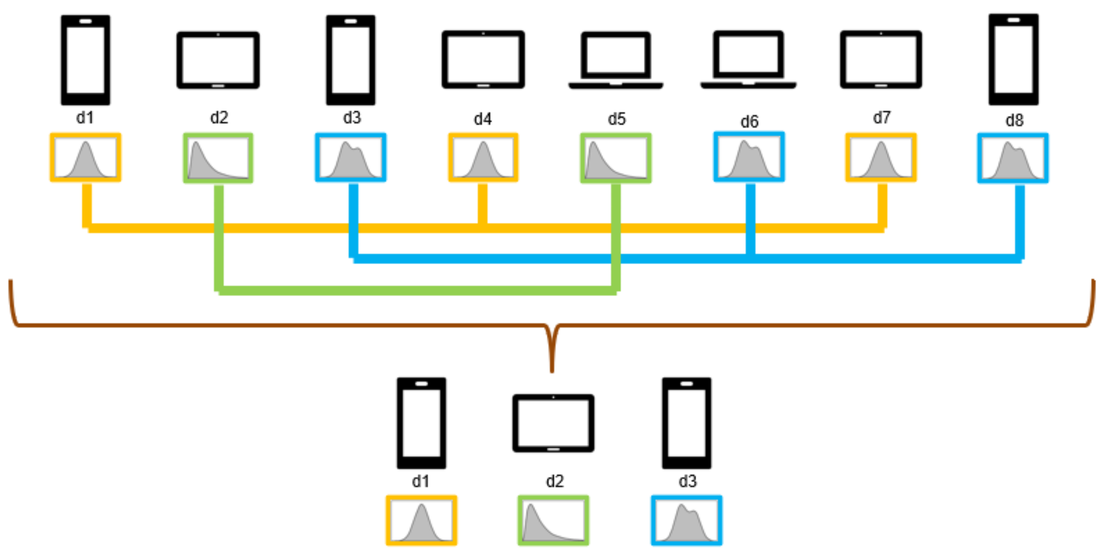
Enforce (ε, 0) – Differential Privacy by adding noise to summaries



System Design



1. **Define a distance metric between device summaries (Hellinger Distance)**
2. **Cluster devices based on their similarity (DBSCAN)**

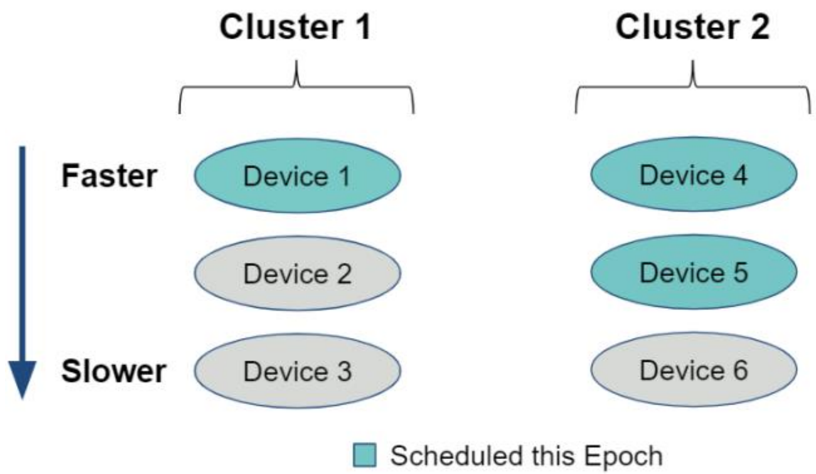


Scheduling Decisions

1. Sort devices within clusters based on performance

2. Assign weights to each cluster using a convex combination of loss and latency reduction

3. Select clusters using weighted random sampling with replacement



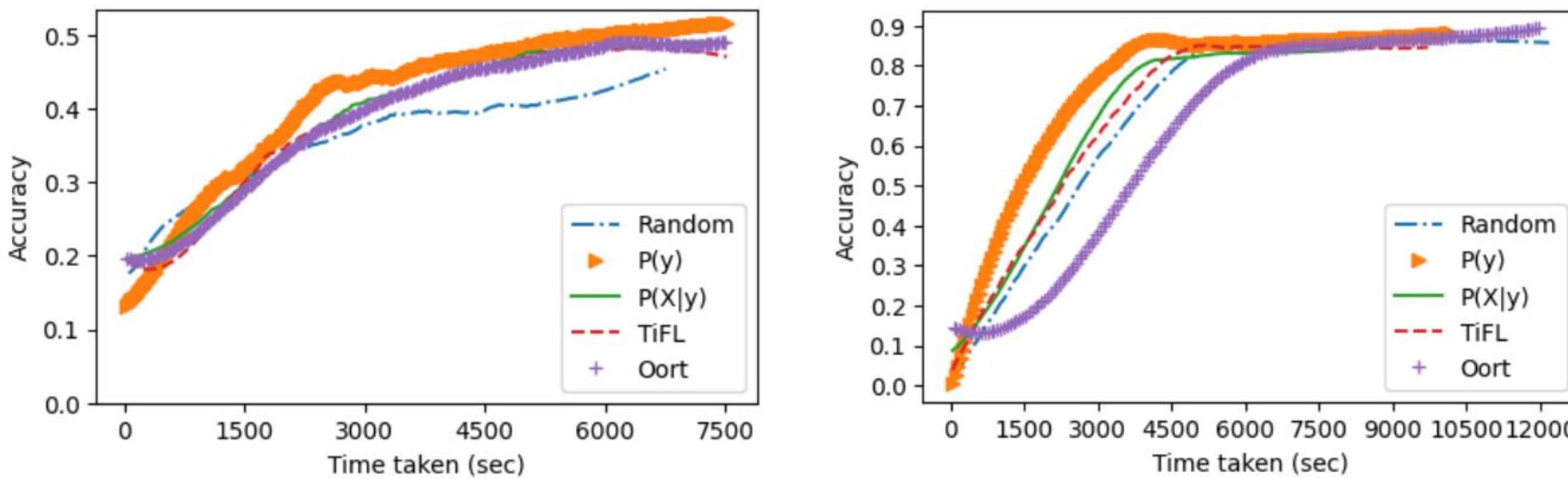
Potential Issue: Summaries only consider part of the joint distribution, which could lead to bias.

Experimental Setup

- 50 simulated devices
  - Delays introduced to simulate network + compute latencies
- Datasets: FEMNIST and CIFAR-10
- Metrics: Time-to-accuracy (TTA) for training a CNN (LeNet)
- Baselines: Random Scheduling, TiFL, and Oort
- Skewed Label Distributions:

Prominent digit (75%)	0	1	2	3	4	5	6	7	8	9
Noise Labels (25%)	2/4/3	0/9/6	7/6/1	8/5/4	0/9/5	3/6/8	2/8/7	4/0/9	2/3/4	1/5/6

Model Convergence

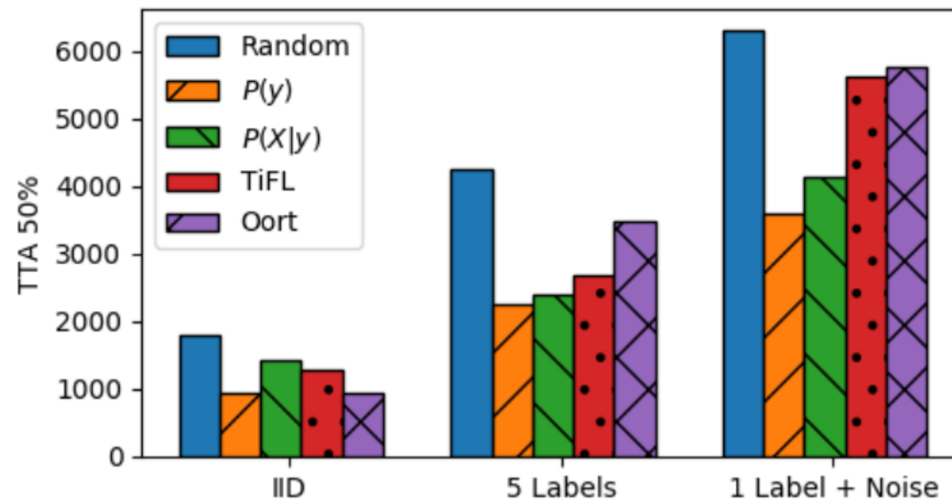


23-27% reduction in training time to reach the same level of accuracy

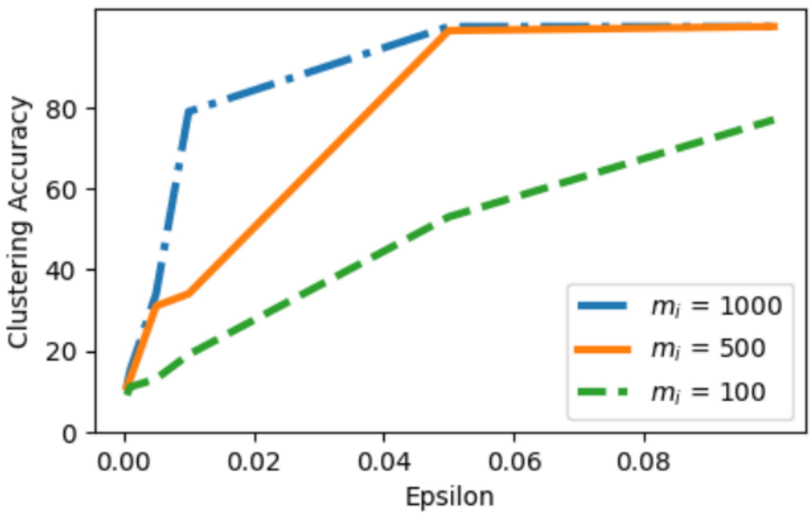
Degrees of Label Skew

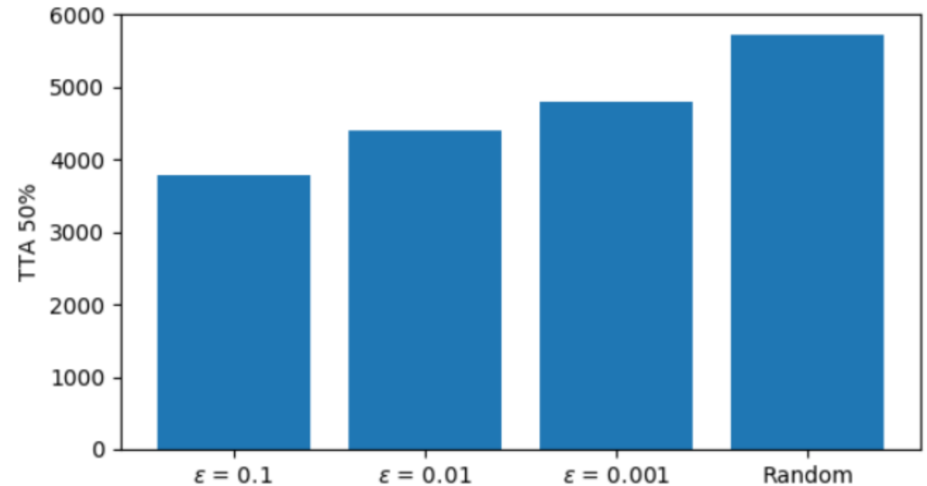
Relative benefit over baselines increases as skew increases

Skew negatively impacts all methods



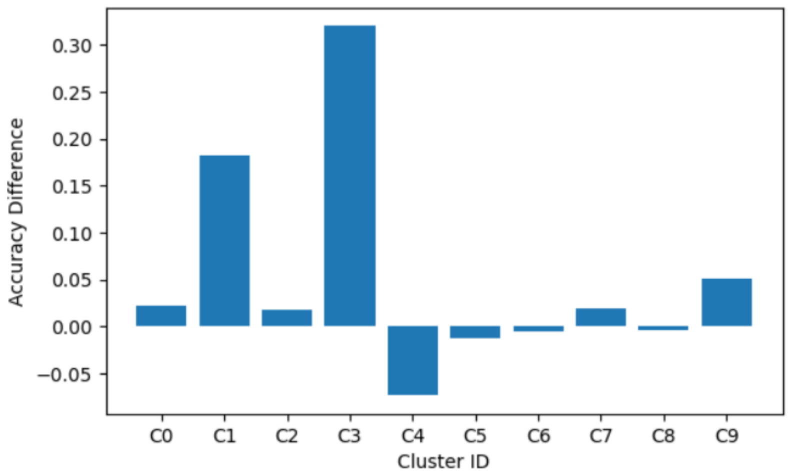
Differential Privacy

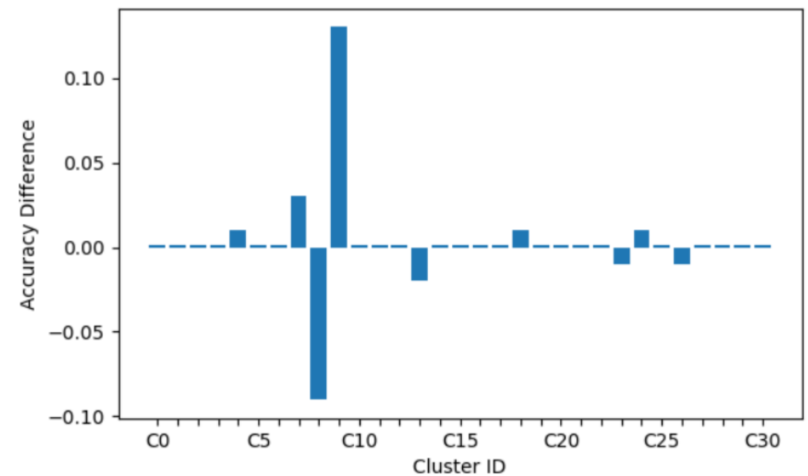




Epsilon parameter can substantially impact clustering performance

Bias Considerations





Some bias observed within p(y) clusters, less with p(x|y)

Conclusion

- We explored the impact of data heterogeneity in federated learning
- Proposed clustering and scheduling methods for mitigating performance degradation
- Observed a 23% to 27% reduction in TTA when leveraging device similarity

Questions?

Distributed Computing Systems Group

UNIVERSITY OF MINNESOTA
Driven to Discover℠