Why is my phone so hot at 4am?

Federated Learning

AMIT SHARMA

JONNY GIORDANO

PRAMIT DUTTA

SHAUNA MOONEY

SUPERVISORS:

DR. SUBHRAKANTI DEY

DR. THOMAS NAUGHTON

DR. EDGAR GALVAN

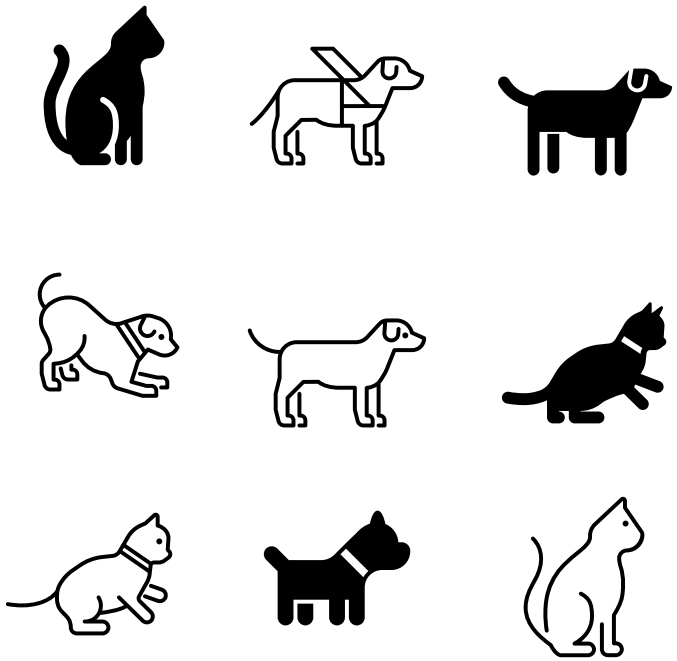
DR. JOHN MCDONALD

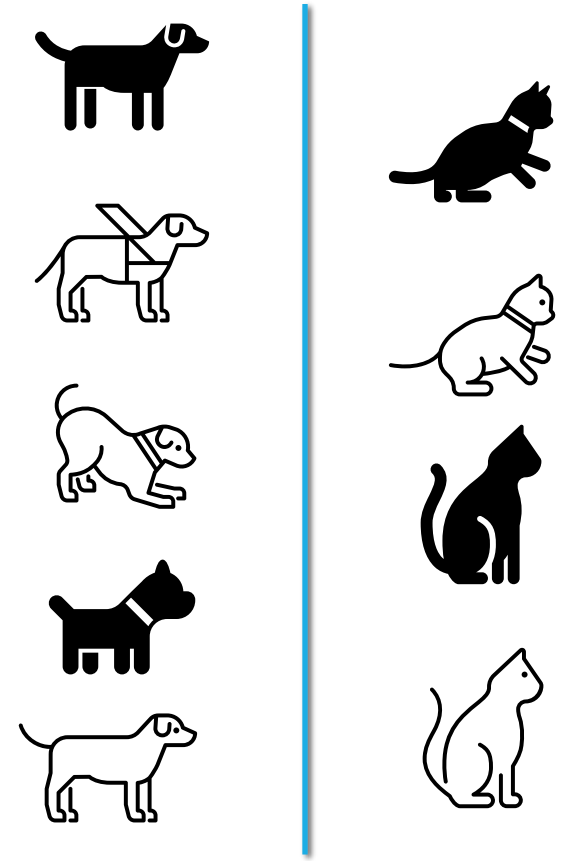
DR. RUDI VILLING

Project Goal

BUILD OUR OWN FEDERATED LEARNING MODEL FOR MOBILE EDGE DEVICES

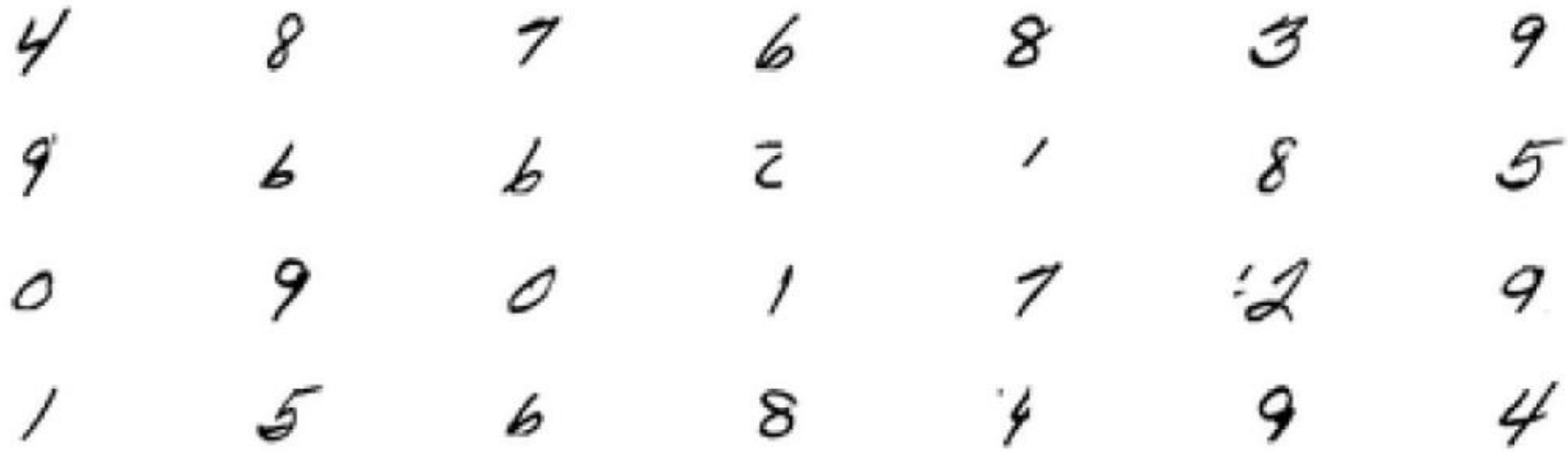
What is Machine Learning?



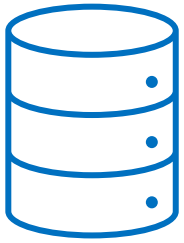


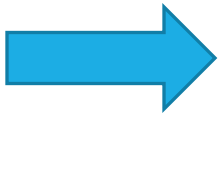
Our Dataset - EMNIST

CLASSIFICATION OF HANDWRITTEN DIGITS 0-9



Traditional Machine Learning

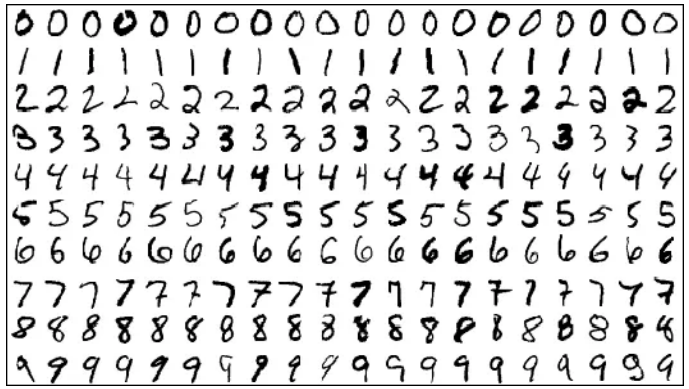




Data Collection

Central Location

Model



Issues

Data must be present at centralized location

Communication Cost

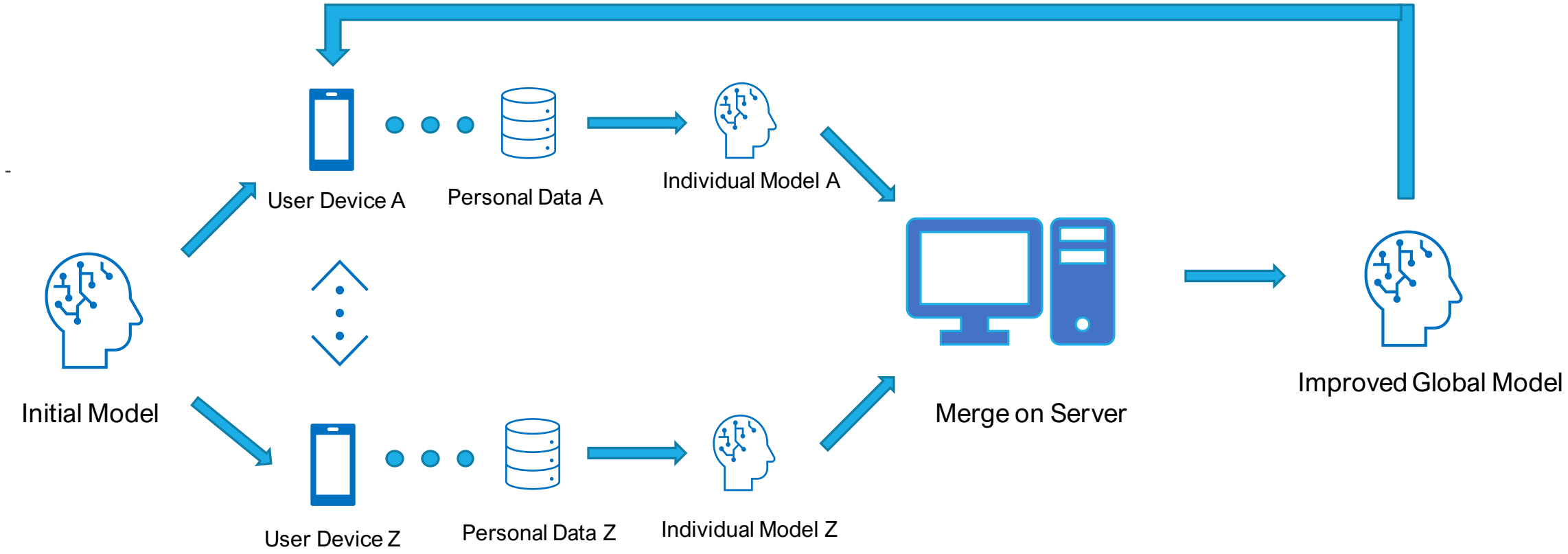
Privacy & Security

Motivation for Federated Learning

Access data without moving it

Leverage power of user devices

Federated Learning (FL)

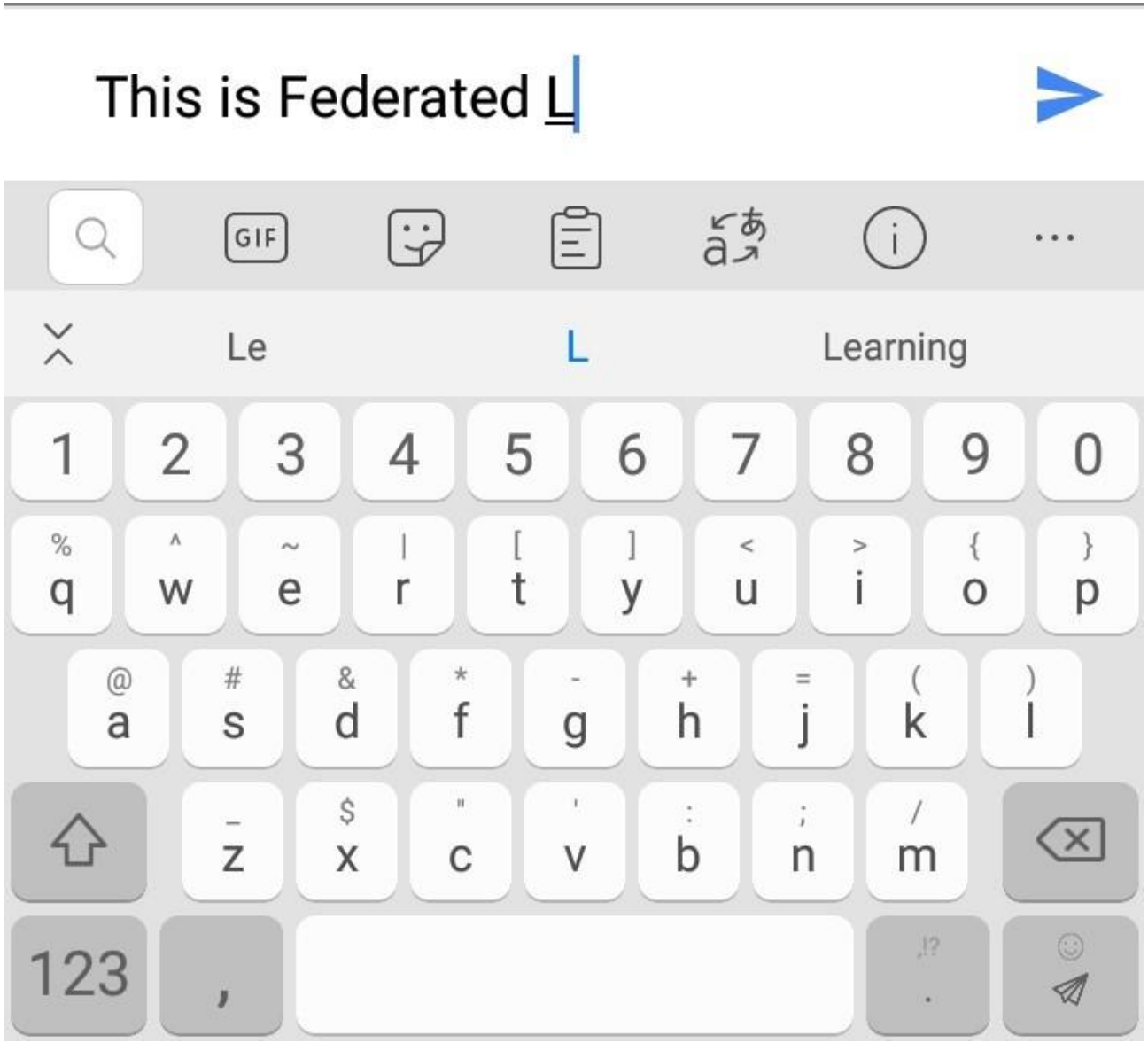


Federated Learning Use Cases

Autocorrect on Smart Phones (GBoard)

Smart Health Devices

Autonomous Vehicles



Why use FL?

1. PRIVACY & SECURITY

2. COMMUNICATION COST

3. LATENCY

Principle Investigations

1

1. **Comparison** of a typical Machine Learning vs Federated Learning

2

2. **Client Selection** for Learning

- **Scheduling Policy**
- Bandwidth Cost

Simple Wireless Communication System

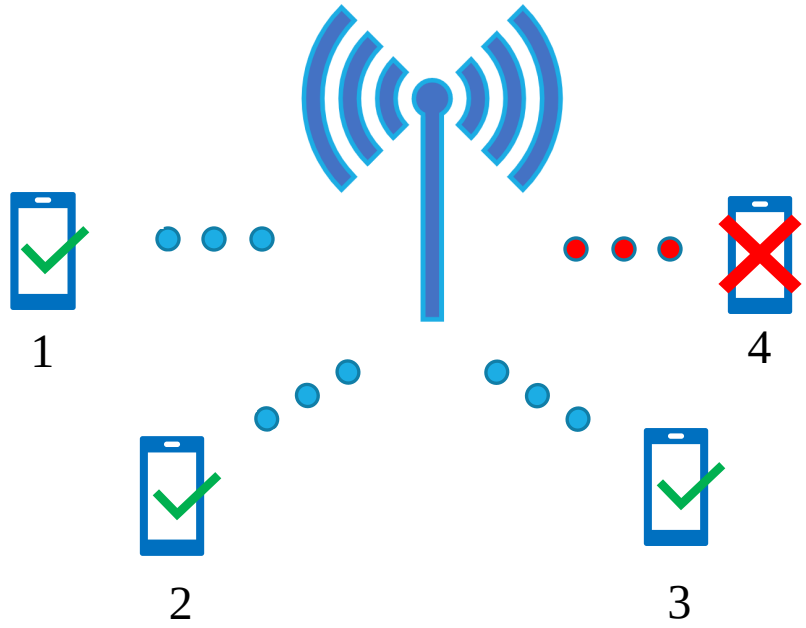
Simple Wireless Communication System

User Equipment

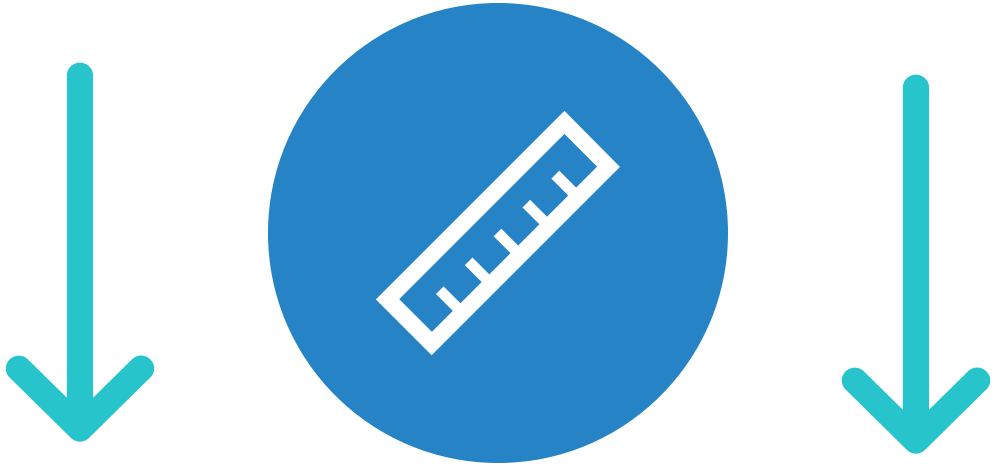
Access Point

Server

Bandwidth Usage



Goal in our FL Model

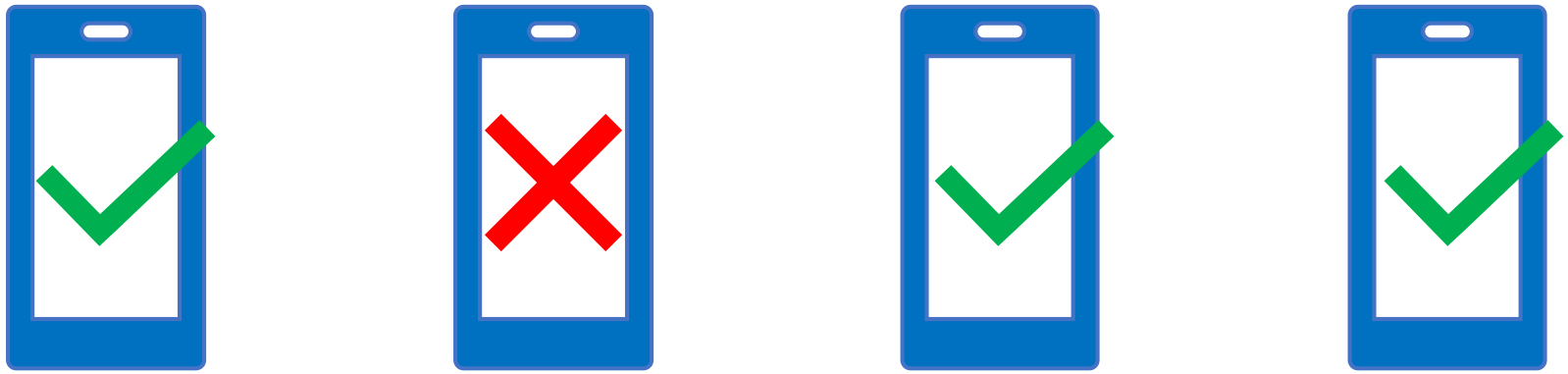


BANDWIDTH USAGE

Lower Bandwidth by Smart Client Selection

Not all clients are equal

A **Scheduling Policy** is a method in which clients are selected



Scheduling Policy – Data Quality

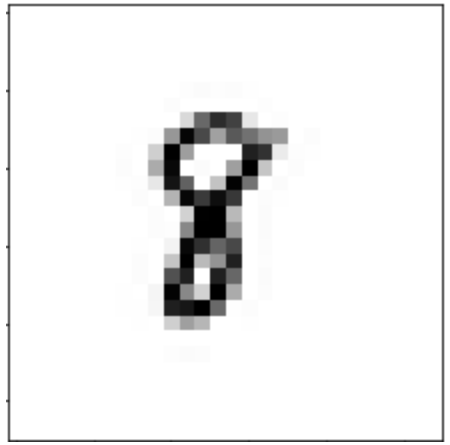
Assign **Client Importance Factor (CIF)** which reflects a client's data quality

Select clients with highest **CIF**

Certain Case

Digits	Probability
0	0.01
1	~0
2	~0
3	~0
4	~0
5	~0
6	~0
7	~0
8	0.96
9	0.02

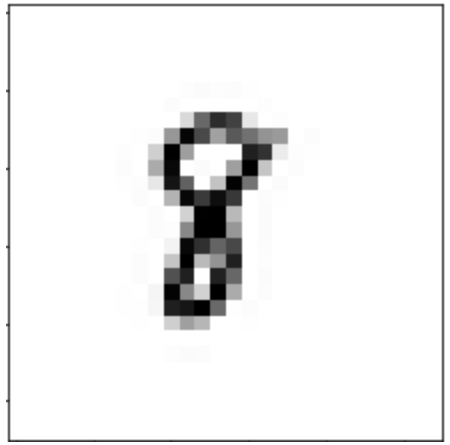
Real Data Sample



Certain Case

Digits	Probability
0	0.01
1	~0
2	~0
3	~0
4	~0
5	~0
6	~0
7	~0
8	0.96
9	0.02

Real Data Sample



(Entropy) $\mathcal{U}_{\mathrm{e}}(\mathbf{x}) = -\sum_{\hat{c}=1}^{C} P_\theta(\hat{c}|\mathbf{x}) \log P_\theta(\hat{c}|\mathbf{x})$

CIF = 0.071

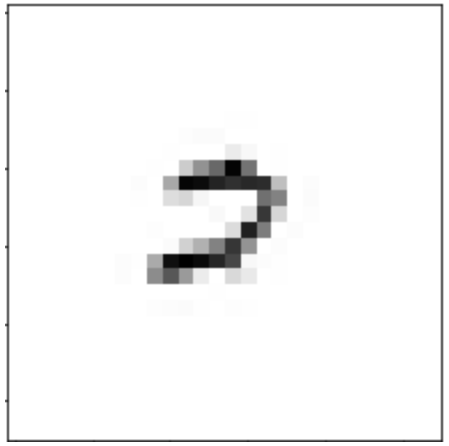
Low Client Importance Factor

Ref: Liu, D., Zhu, G., Zhang, J., & Huang, K. (2020). Data-importance aware user scheduling for communication-efficient edge machine learning. IEEE Transactions on Cognitive Communications and Networking.

Uncertain Case

Digits	Probability
0	0.04
1	~0
2	0.34
3	~0.31
4	~0
5	~0
6	~0
7	~0.30
8	~0
9	~0

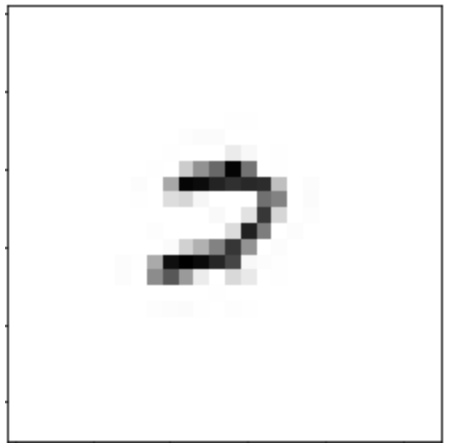
Real Data Sample



Uncertain Case

Digits	Probability
0	0.04
1	~0
2	0.34
3	~0.31
4	~0
5	~0
6	~0
7	~0.30
8	~0
9	~0

Real Data Sample



$$\textbf{(Entropy)} \quad \mathcal{U}_{\mathrm{e}}(\mathbf{x}) = -\sum_{\hat{c}=1}^{C} P_\theta(\hat{c}|\mathbf{x}) \log P_\theta(\hat{c}|\mathbf{x})$$

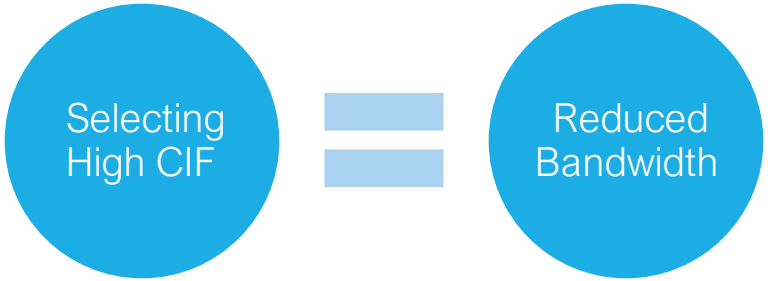
CIF = 0.523

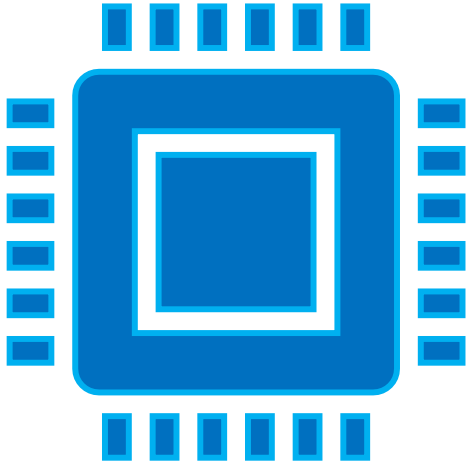
High Client Importance Factor

Ref: Liu, D., Zhu, G., Zhang, J., & Huang, K. (2020). Data-importance aware user scheduling for communication-efficient edge machine learning. IEEE Transactions on Cognitive Communications and Networking.

Recap

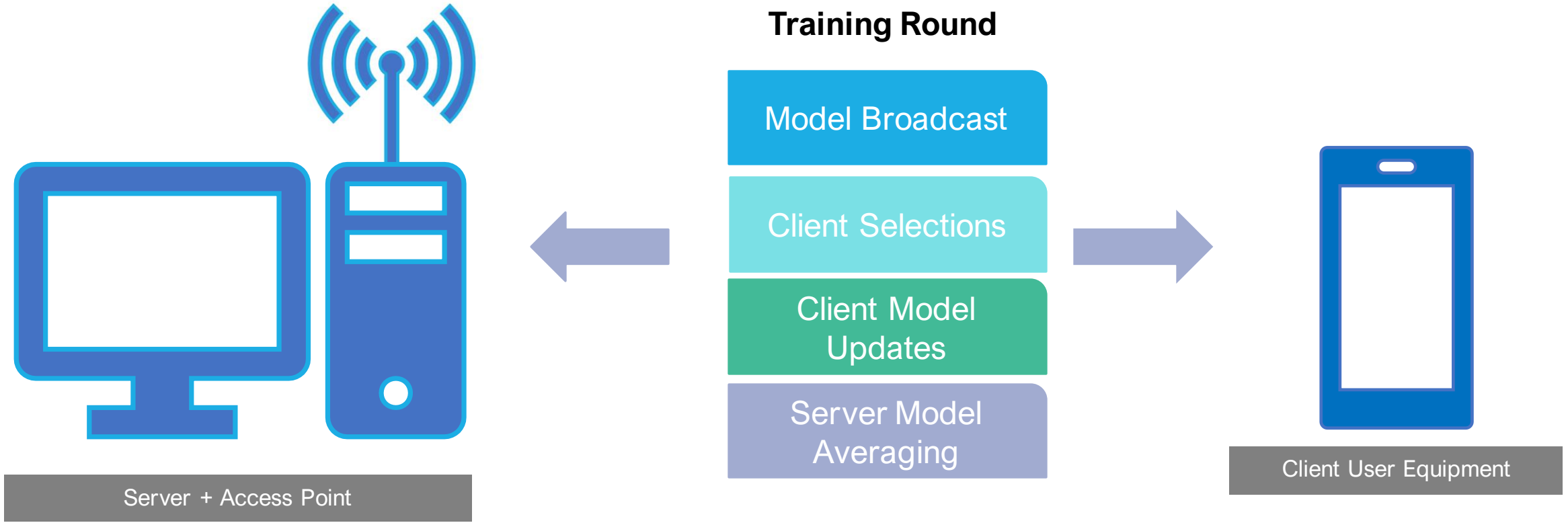
1) Bandwidth consumption must be minimized in FL, so communication is not clogged
2) Bandwidth overall can be conserved by reducing the number of learning rounds
3) Learning rounds can be reduced by selecting clients with high CIF ("good" data)





Our Simulation

Our Simulation



Server-Client Simulation

Federated Averaging Algorithm

- Core of Federated Learning Technique
- Single layer NN
- Computes the "weighted average" of the "Model Weights and Biases" of selected clients in each round
- *K* clients are indexed by *k*
- *B* is the local minibatch size
- *E* is the number of local epochs , fixed at 10 per round
- η is the client learning rate = 0.01
- Optimizer used is SGD

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

Ref: H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, Blaise Aguera y Arcas. Communication-Efficient Learning of Deep Networks from Decentralized Data. Proceedings of the 20 the International Conference on Artificial Intelligence and Statistics (AISTATS) 2017. JMLR: W&CP volume 54

Server and Client Datasets

- We use TFF - EMNIST - Non-IID Dataset
- Dataset provided as Train and Test data for individual clients defined by Client-IDs , Max 3382 Clients
- Our Implementation:

SERVER

Train Dataset:
Concatenated Data of
Client-IDs: 2000-2500

Test Dataset:
Concatenated Data of
Client-IDs: 80-500

CLIENT

Train Dataset:
Individual Data from
Client-IDs: 0-100

Validation Dataset:
Validation Split of 0.2

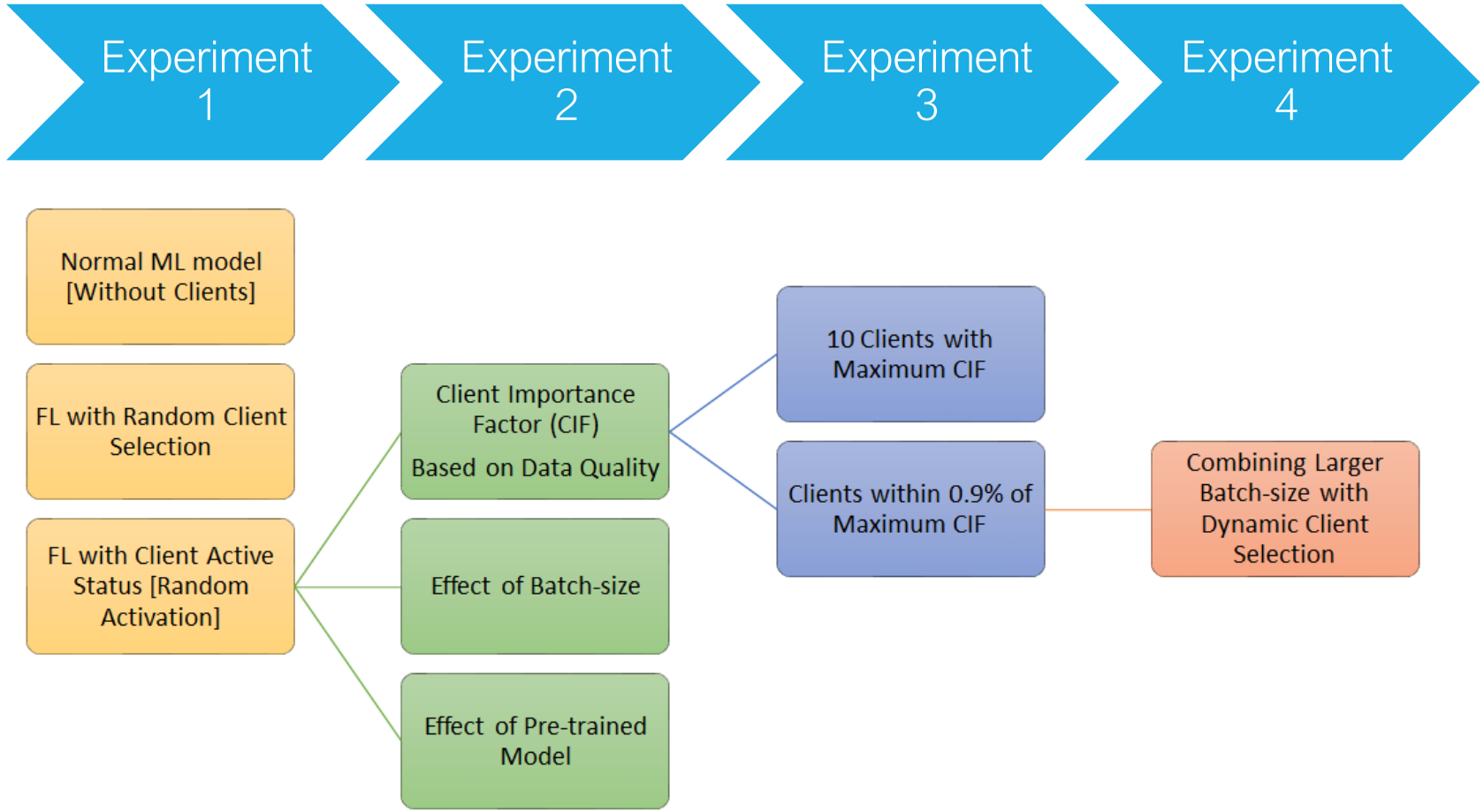
Ref: LEAF: A Benchmark for Federated Settings

Packages

- Python
- Numpy for data manipulation
- TFF (TensorFlow Federated) for using EMNIST dataset
- TensorFlow & Keras for building neural network architecture
- Matplotlib for plots

Client Selection Criteria [Experiments]

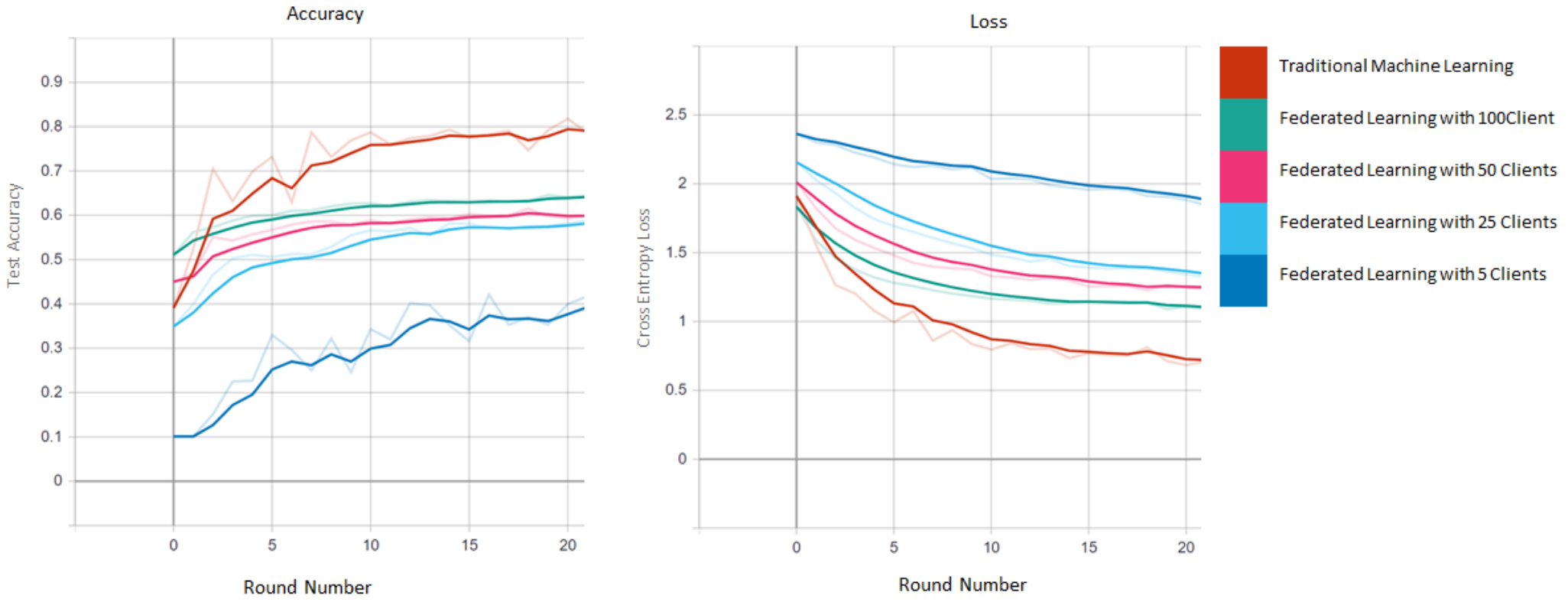
Various Client selection criterions are evaluated within this project to understand the impact on the model training accuracy



Our Entire Code and Results are Available on : https://github.com/pramitd/Federated_Learning

Results

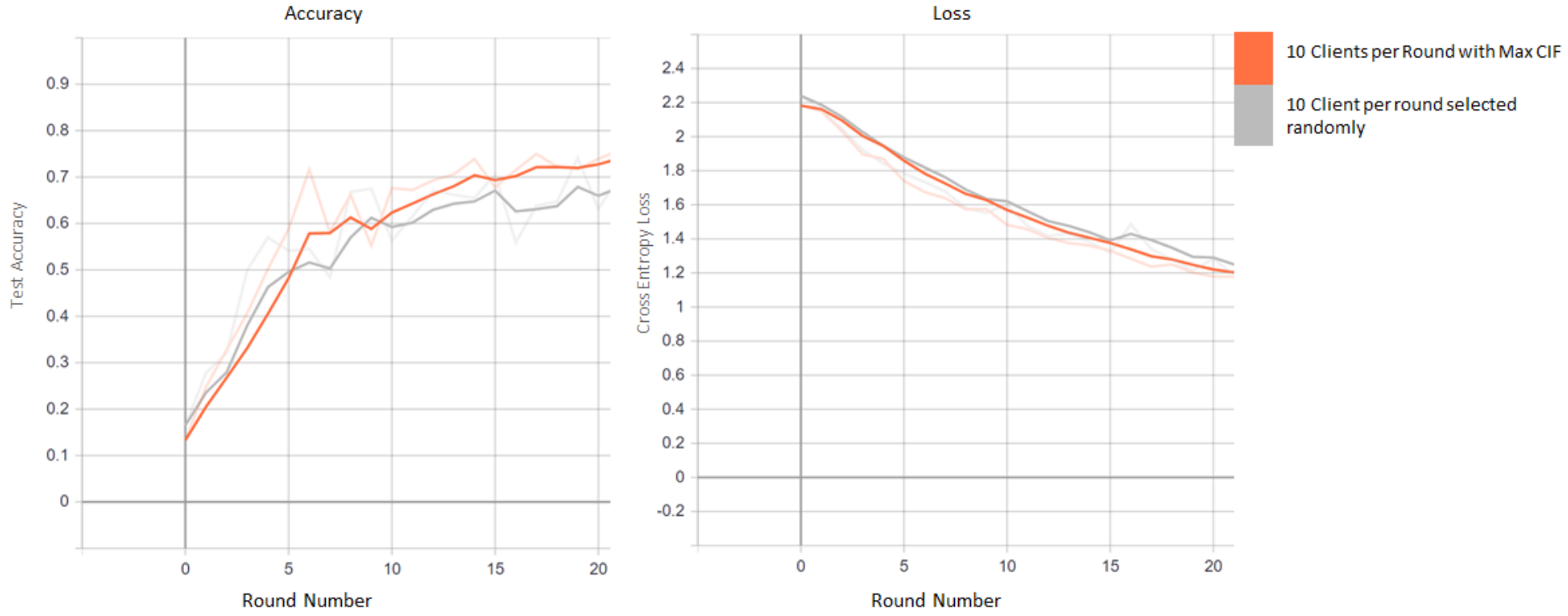
Federated Learning vs Traditional Machine Learning



Training Features

- Same Dataset for Clients and Server
- Tensor-board Smoothing of 0.6
- Model weights and biases initialized to zero
- Data Batch size per round = 20

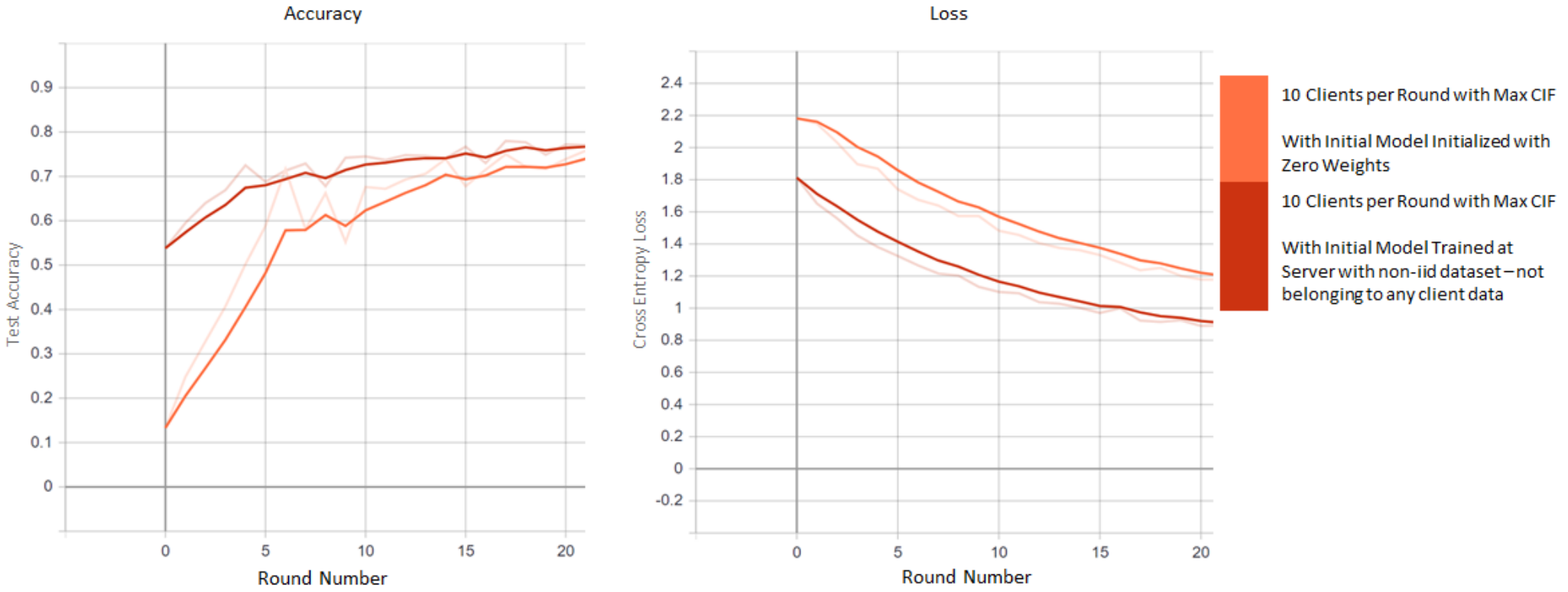
Scheduling Policies - Client Selection with CIF vs Random Client Selection



Training Features

Same Client Pool Used (Client ID 0-100)	Tensor-board Smoothing of 0.6	Total # Devices used to achieve Maximum Accuracy of ~**72% = 200**

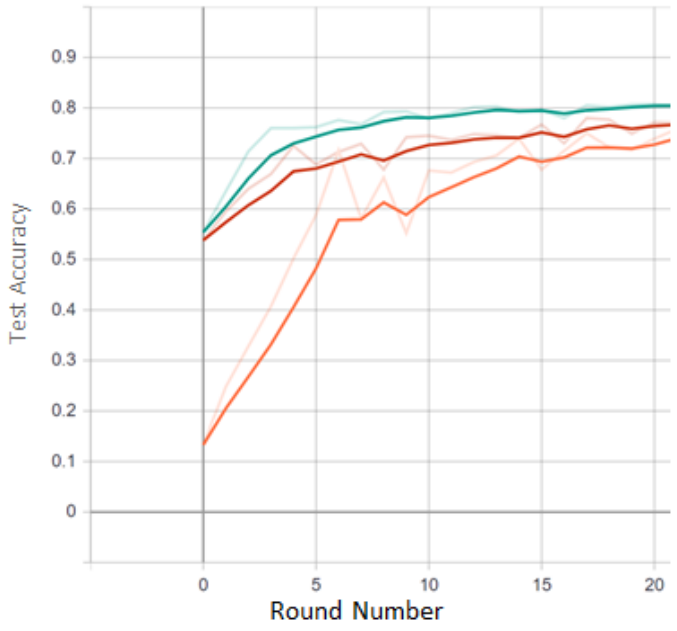
Naïve Model vs Pre-Trained Model

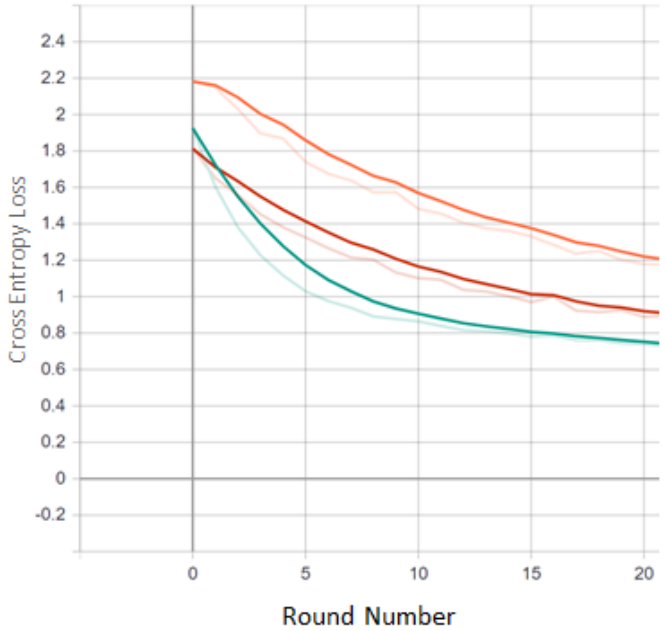


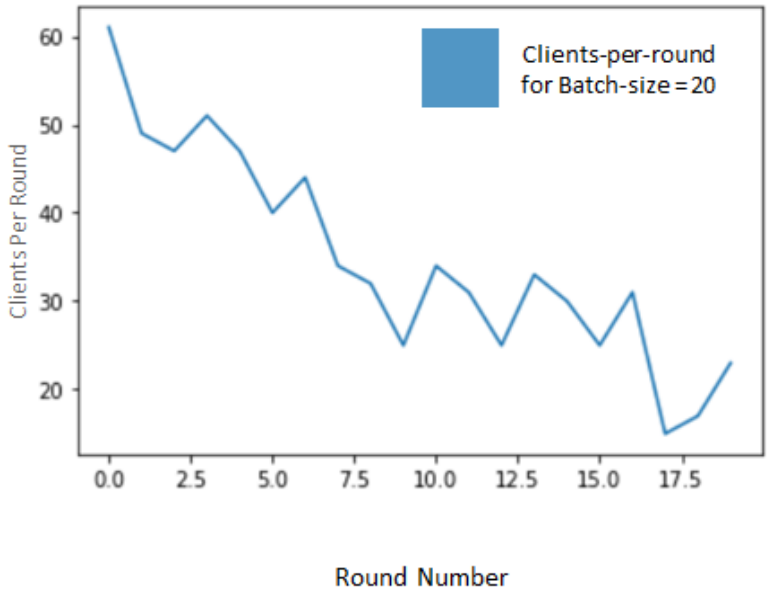
Training Features

Same Client Pool Used (Client ID 0-100)	Tensor-board Smoothing of 0.6	Total # Devices used to achieve Maximum Accuracy of ~**76.4%** = **200**

Scheduling Policies - Fixed Client Selection vs Dynamic Client Selection







10 Clients per Round with Max CIF
With Initial Model Initialized with **Zero Weights**

10 Clients per Round with Max CIF
With Initial Model **Pre-Trained** at Server with non-IID dataset – not belonging to any client data

Clients Selected with CIF within 10% of Maximum CIF Value
With Initial Model Initialized with **Zero Weights**

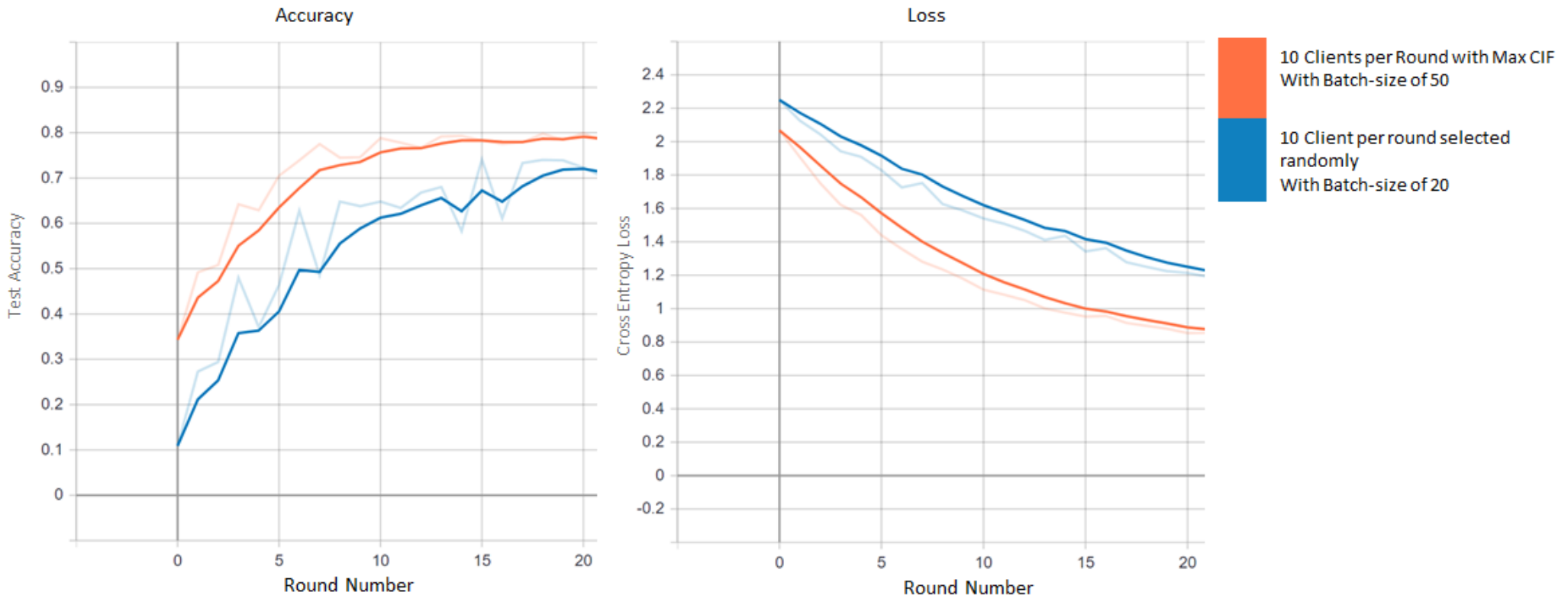
Training Features

Same Client Pool Used (Client ID 0-100)

Tensor-board Smoothing of 0.6

Total # Devices used to achieve Maximum Accuracy of ~80% =672

Effect of Batch-size per round of Client Data



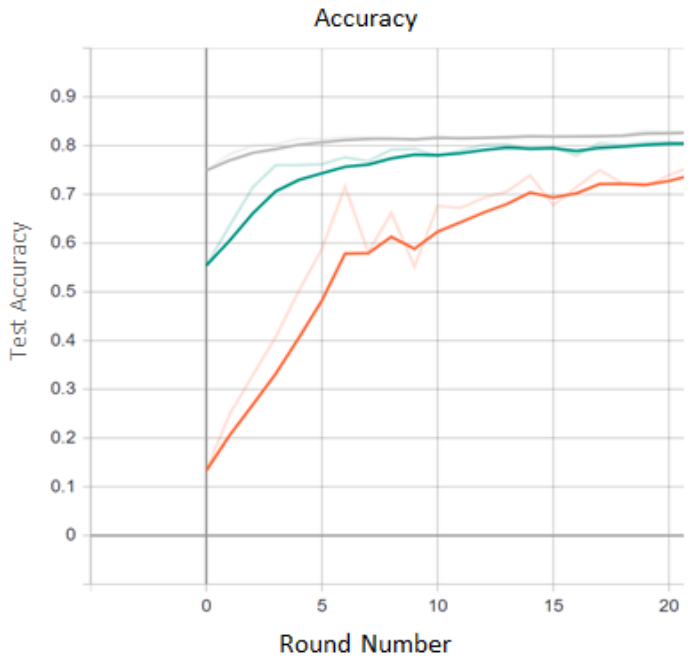
Training Features

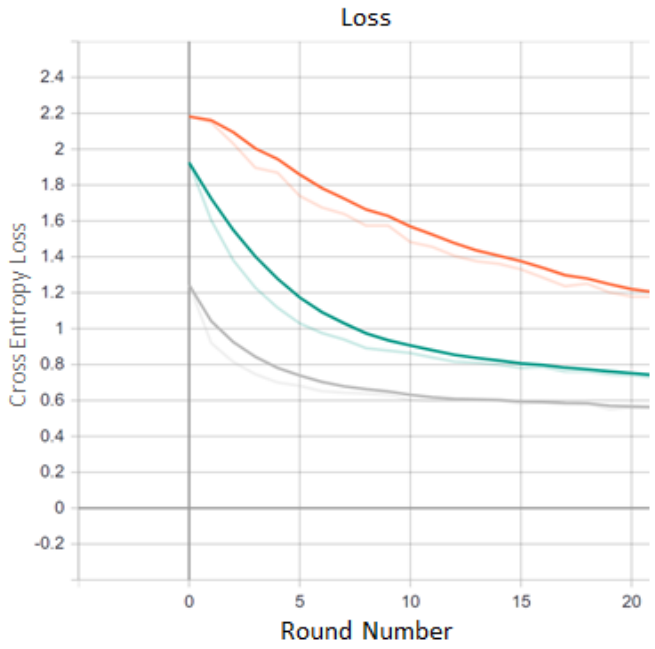
Same Client Pool Used
(Client ID 0-100)

Tensor-board
Smoothing of 0.6

Total # Devices used to
achieve Maximum Accuracy of **~78.1% = 200**

Combining Batch-size with Dynamic Client Selection





10 Clients per Round with Max CIF

Clients Selected with CIF within 10% of Maximum CIF Value With Batch-size = 20

Clients Selected with CIF within 10% of Maximum CIF Value With Batch-size = 50

Training Features

Same Client Pool Used (Client ID 0-100)

Tensor-board Smoothing of 0.6

Total # Devices used to achieve Maximum Accuracy of **~80% = 184**

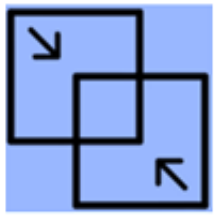
Conclusions

1. Federated Learning (FL) is a new paradigm in

Machine Learning (ML) that enables security and privacy of user data.

2. FL is slower in convergence than traditional centralized ML, however, smarter client selection can improve on the same.

3. Smart Client selection can reduce the communication cost and depends largely on client data quality and client hardware.



4. Combining multiple methods of Smart Client selection strategies give the best results.

Why your phone is so hot at 4am?

YOUR PHONE HAS BEEN WORKING HARD,

CONTRIBUTING TO BUILD A SMARTER MACHINE LEARNING MODEL BASED ON YOUR DATA,

MAINTAINING THE SECURITY AND PRIVACY OF YOUR DATA

FEDERATED LEARNING

Thanks for Watching

QUESTIONS?

Supplementary Slides

Entropy – Calculating Uncertainty

$$\textbf{(Entropy)} \quad \mathcal{U}_{\mathrm{e}}(\mathbf{x}) = -\sum_{\hat{c}=1}^{C} P_\theta(\hat{c}|\mathbf{x}) \log P_\theta(\hat{c}|\mathbf{x})$$

Where **ĉ** is a possible prediction
and **x** is a data point

Liu, D., Zhu, G., Zhang, J., & Huang, K. (2020). *Data-Importance Aware User Scheduling for Communication-Efficient Edge Machine Learning. IEEE Transactions on Cognitive Communications and Networking, 1–1.* doi:10.1109/tccn.2020.2999606

Proportional Fair - SNR

$$\mathbf{m}^* = \underset{\mathbf{m} \subset \{1,2,\ldots,K\}}{\arg\max} \left\{ \frac{\tilde{\rho}_{m_1,t}}{\bar{\rho}_{m_1,t}}, \ldots, \frac{\tilde{\rho}_{m_N,t}}{\bar{\rho}_{m_N,t}} \right\}$$

Where *K* is number of UE's
and *m* is a N-length vector of UEs
and $\tilde{\rho}_{m_i,t}$ is the SNR of a UE on a given round
and $\bar{\rho}_{m_i,t}$ is the average SNR of a UE

Yang, H. H., Liu, Z., Quek, T. Q., & Poor, H. V. (2019). Scheduling policies for federated learning in wireless networks. *IEEE Transactions on Communications*, *68*(1), 317-333.

Data Quality Measurement

Data Quality is measured using the concept of data entropy:

$$\xi(x) = -\sum_{\hat{c}}^{C} P_\theta(\hat{c} \mid x) * \log P_\theta(\hat{c} \mid x)$$

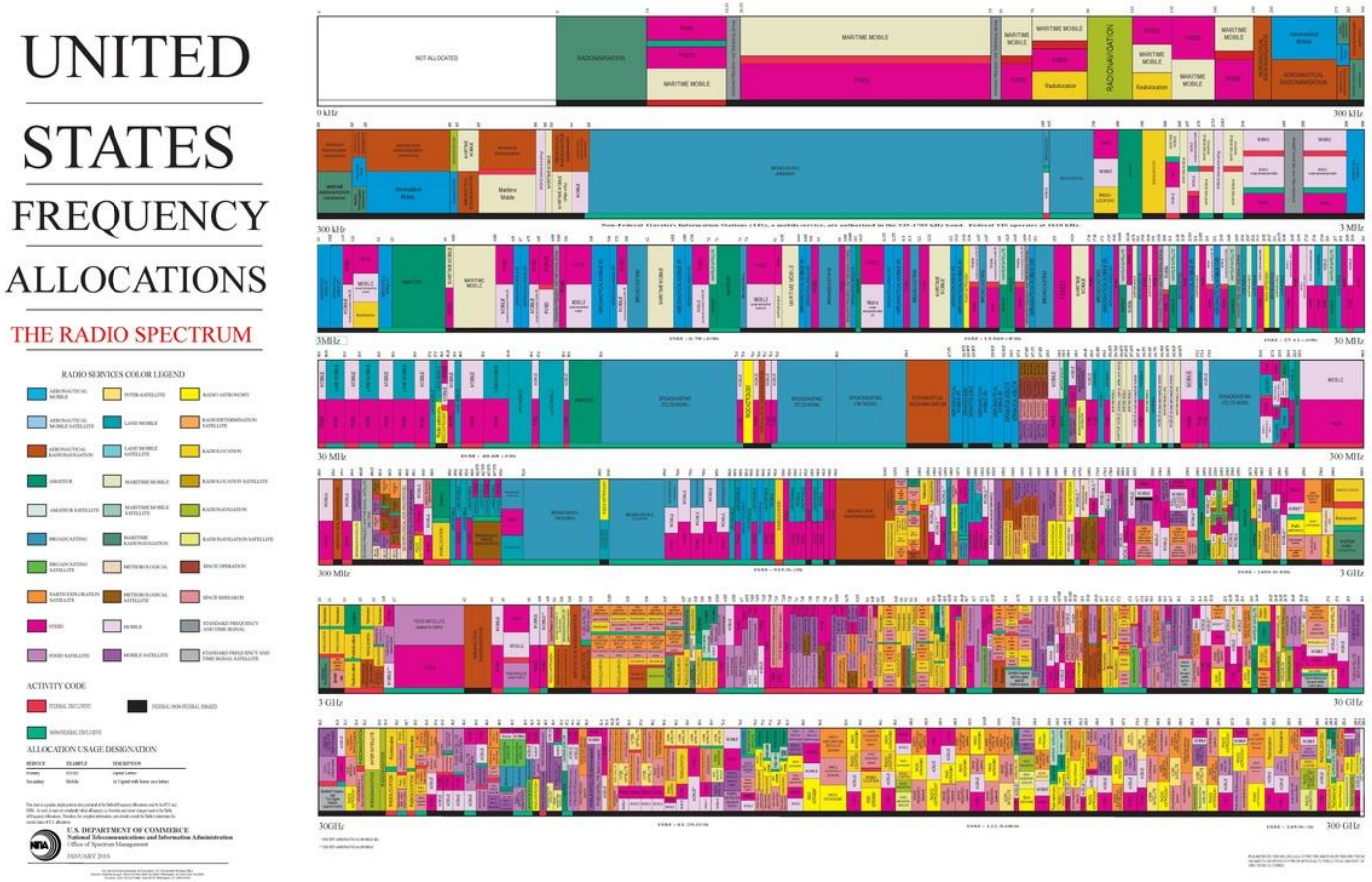
$P_\theta(\hat{c} \mid x)$ = Class Probabilities of given model with parameters (θ) for a given sample x

For a given set of new client (*k*) dataset the Data-quality index (*I*) is thus defined as:

$$I_k = max_{n \epsilon N}\left[\xi(x_{k,n})\right]$$

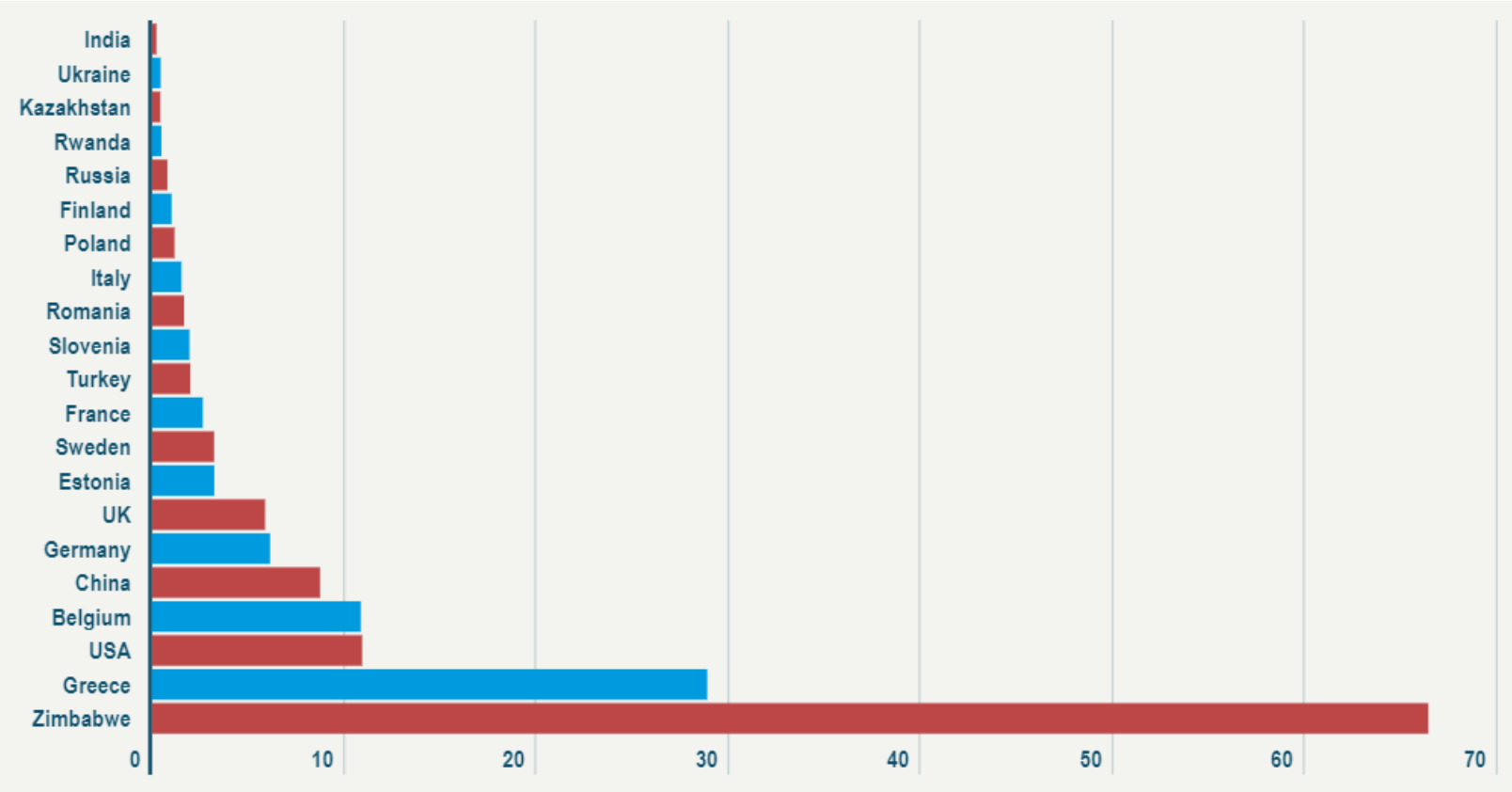
Once the Data-quality index is known for all available clients , we use various hyper-parameters and selection methods to get model updates and perform averaging

Ref:



Spectrum Scarcity

COST OF 1GB DATA IN EUROS



Data Cost

$EMA = \text{Acc}(t) \times k + EMA(t\text{-}1) \times (1 - k)$

where:

t = current time point

$t\text{-}1$ = previous time point

$k = 2 \div (N + 1)$

N = number of time point in EMA

Smoothing in Tensor-board