PRIVACY-PRESERVING LEARNING
FEDERATED LEARNING

Alberto Archetti
Politecnico di Milano
18 Feb. 2022
alberto.archetti@polito.it

DATA ISLANDS

Data is born at the edge

Pros of processing directly at the edge:

- Low latency
- Communication
- Energy efficiency
- Privacy

GDPR and privacy regulation laws

P. Kairouz *et al.*, "Advances and Open Problems in Federated Learning," *arXiv:1912.04977 [cs, stat]*, Mar. 2021

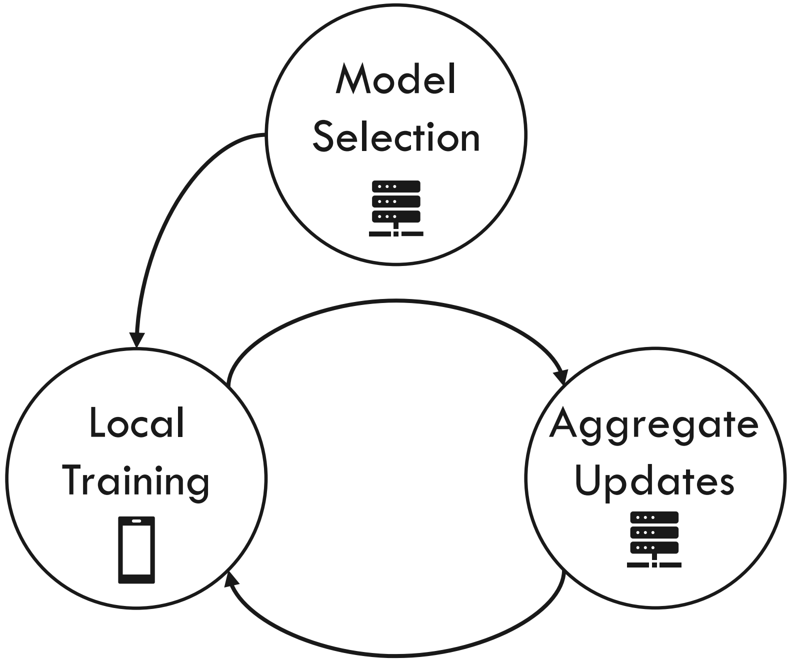
FEDERATED LEARNING (FL)

"Federated learning is a machine learning setting where multiple entities (clients) collaborate in solving a machine learning problem, under the coordination of a central server or service provider.

Each client's raw data is stored locally and not exchanged or transferred; instead, focused updates intended for immediate aggregation are used to achieve the learning objective."

P. Kairouz *et al.*, "Advances and Open Problems in Federated Learning," *arXiv:1912.04977 [cs, stat]*, Mar. 2021

THE GENERAL FL PIPELINE



Model Selection (server)

Define and initialize a global ML model, then send it to the clients

Local Training (clients)

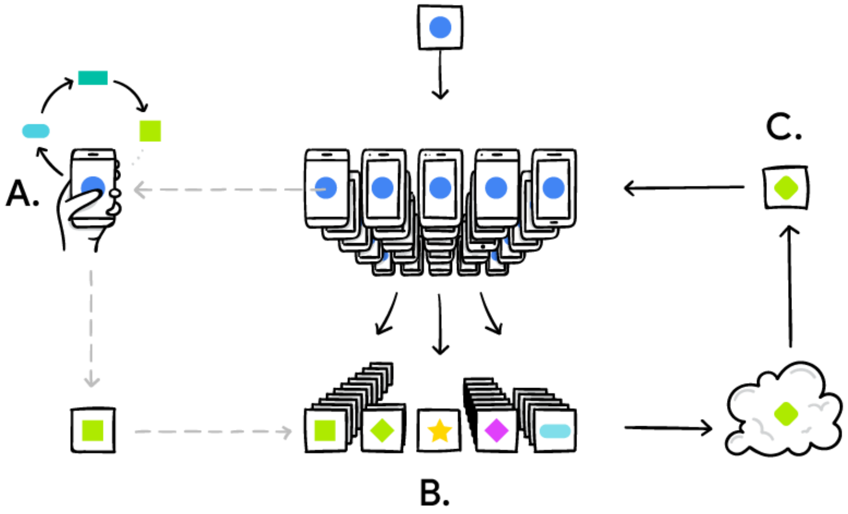
Train the global model on private data, then send the updated model back to the server

Aggregate Updates (server)

Combine the local updates into a single, new, global model, then repeat the process

A. Hard *et al.*, "Federated Learning for Mobile Keyboard Prediction," *arXiv:1811.03604 [cs]*, Feb. 2019

FL AND THE GOOGLE KEYBOARD



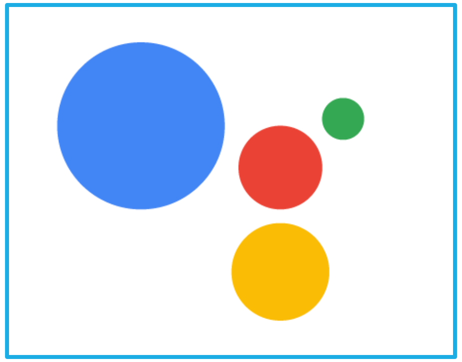
A. Each client computes a step of stochastic gradient descent locally on private data

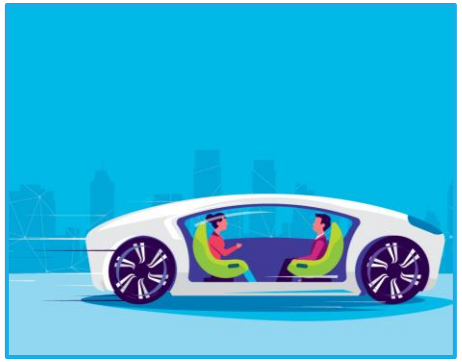
B. The server collects the gradients and performs an aggregated update on the previous model

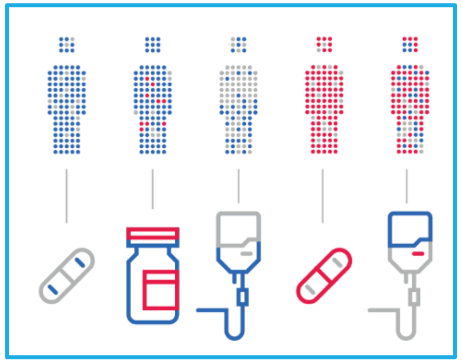
C. The new model is broadcasted to the clients and the process repeats

A. Hard *et al.*, "Federated Learning for Mobile Keyboard Prediction," *arXiv:1811.03604 [cs]*, Feb. 2019

EXAMPLE APPLICATIONS







Voice recognition and vocal assistants on smartphones and embedded devices

Smart adaptation to a dynamic environment in autonomous vehicles

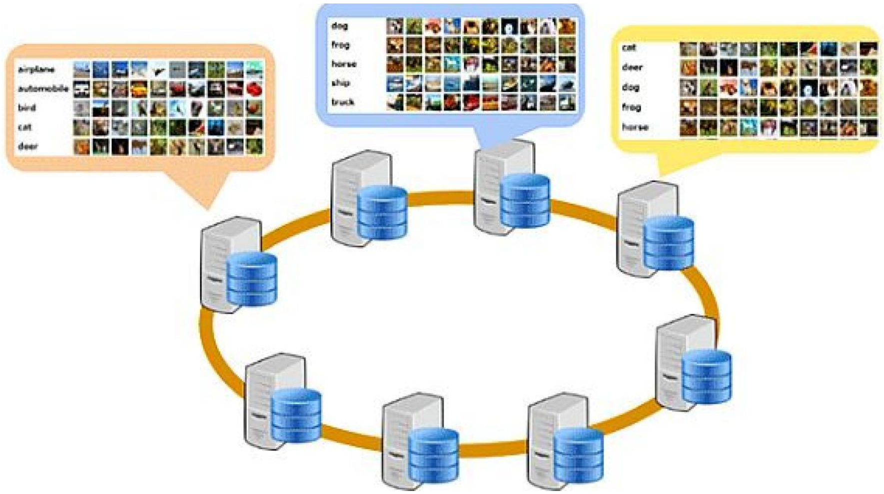
Personalized healthcare on wearable devices

Predictive maintenance in industry

Smart cities

T. Li, "Federated Optimization in Heterogeneous Networks," *talk at On-device Intelligence Workshop, MLSys*, 2020

A NEW PARADIGM



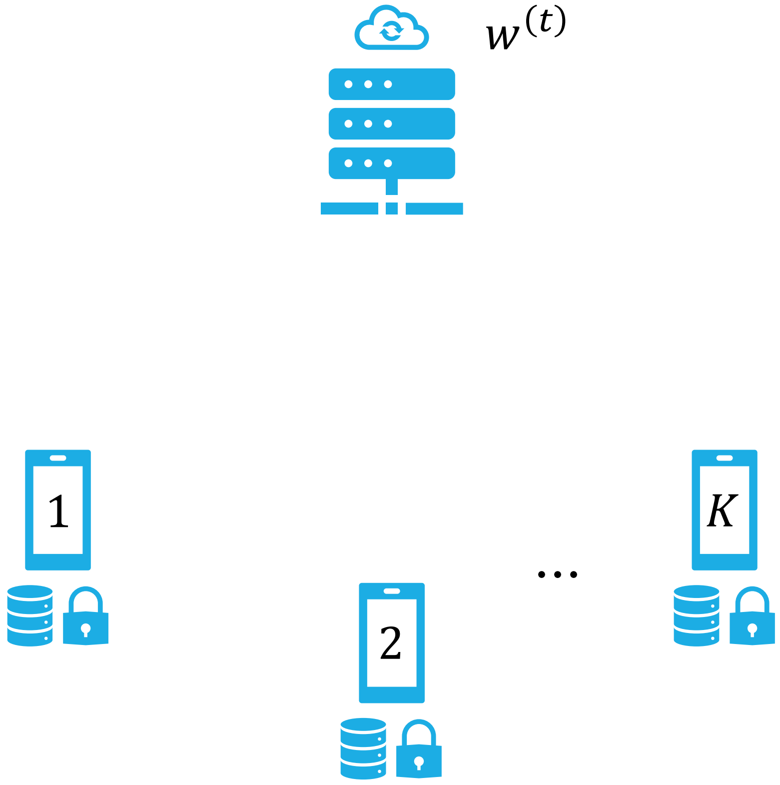
FL is fundamentally different from distributed machine learning, where:

- Data are stored in a network of powerful cloud machines
- Data can be shuffled and balanced across clients
- Any client has access to any part of the dataset
- Computation is the bottleneck
- Typically, 1-1000 clients

A. Willing, "Asynchronous distributed deep learning technology," *eenewseurope.com*, Aug. 2020

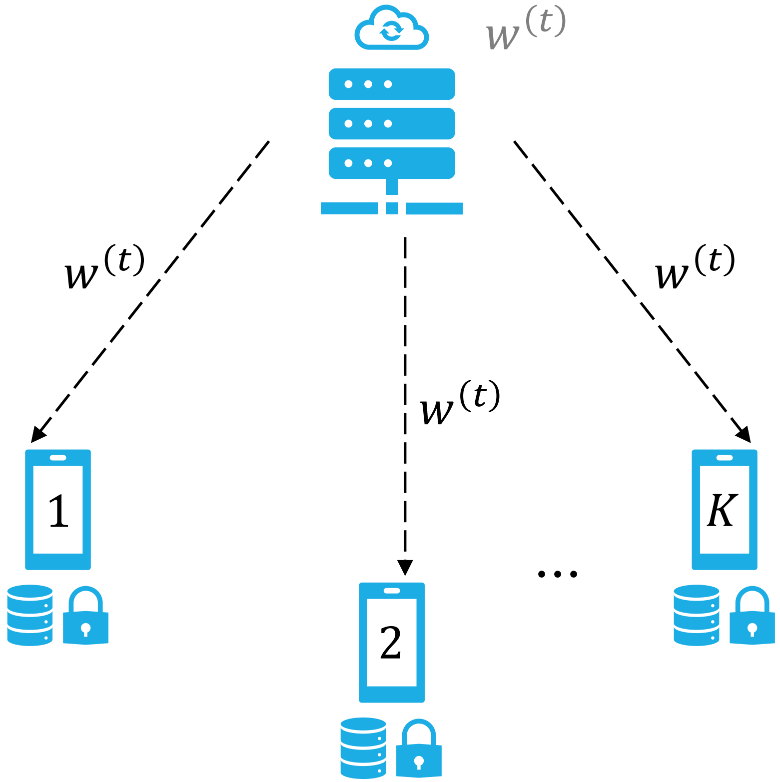
OUTLINE

FEDERATED AVERAGING



1. Select a random set of K clients

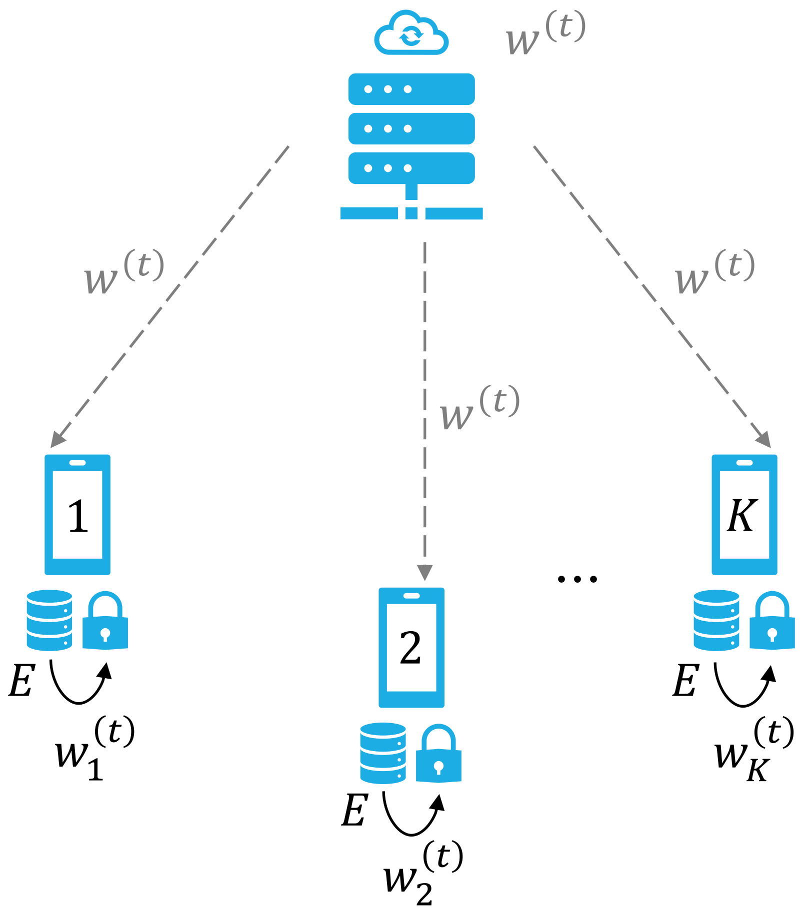
FEDERATED AVERAGING



1. Select a random set of K clients
2. Broadcast $w^{(t)}$

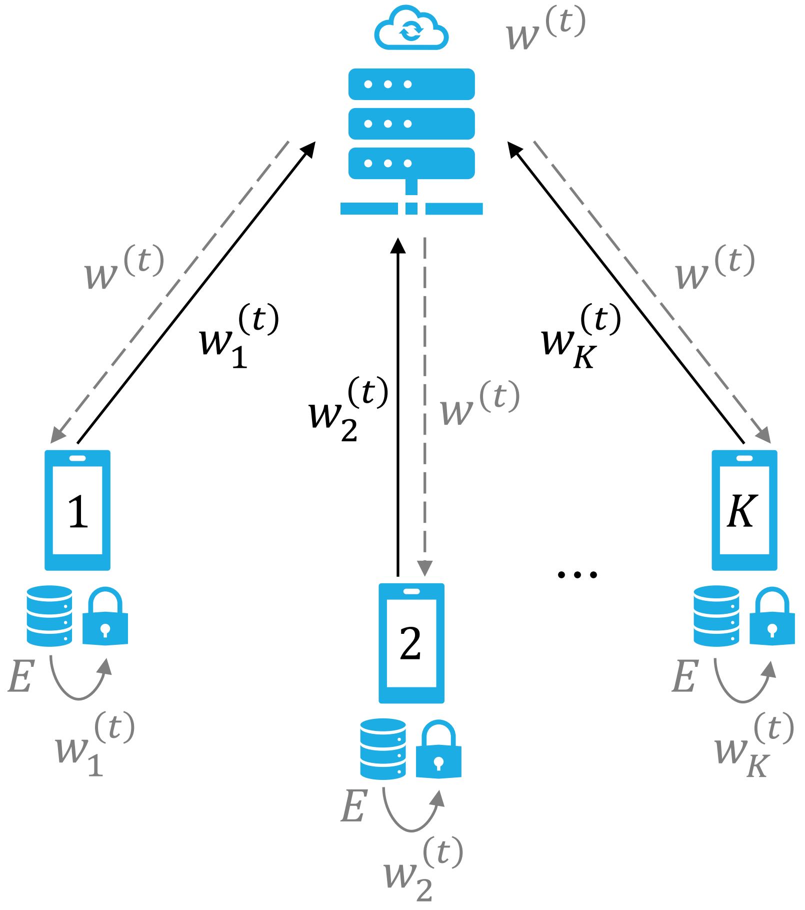
H. B. McMahan *et al.*, "Communication-Efficient Learning of Deep Networks from Decentralized Data," *arXiv:1602.05629 [cs]*, Feb. 2017

FEDERATED AVERAGING



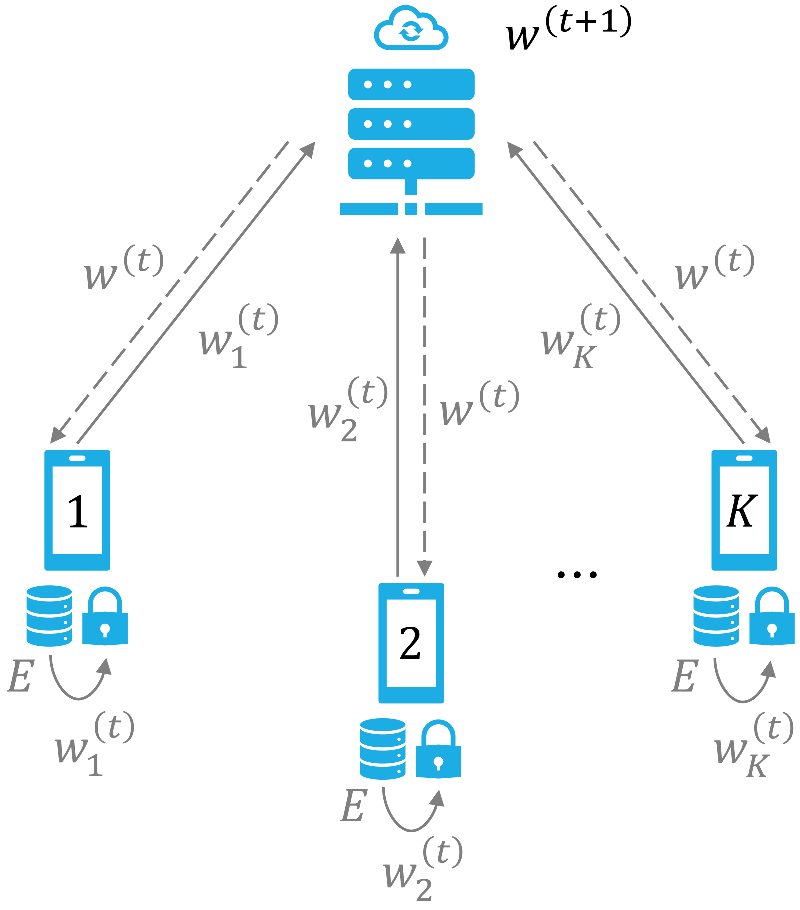
1. Select a random set of K clients
2. Broadcast $w^{(t)}$
3. Perform E iterations of SGD locally as $w_k^{(t)} \leftarrow w_k - \eta \nabla \mathcal{L}(w; b)$

FEDERATED AVERAGING



1. Select a random set of K clients
2. Broadcast $w^{(t)}$
3. Perform E iterations of SGD locally as $w_k^{(t)} \leftarrow w_k - \eta \nabla \mathcal{L}(w; b)$
4. Send $w_k^{(t)}$ back to the server

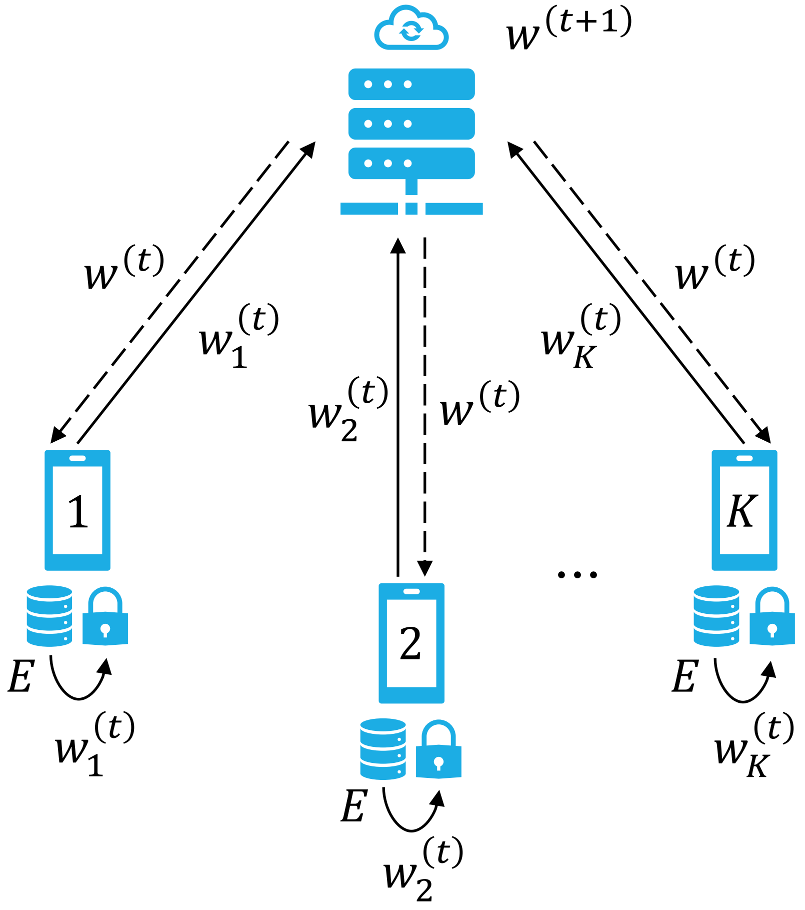
FEDERATED AVERAGING



1. Select a random set of K clients
2. Broadcast $w^{(t)}$
3. Perform E iterations of SGD locally as $w_k^{(t)} \leftarrow w_k - \eta\nabla\mathcal{L}(w; b)$
4. Send $w_k^{(t)}$ back to the server
5. Aggregate updates as $w^{(t+1)} \leftarrow \sum_{k=1}^{K} \frac{n_k}{n} w_k^{(t)}$

H. B. McMahan *et al.*, "Communication-Efficient Learning of Deep Networks from Decentralized Data," *arXiv:1602.05629 [cs]*, Feb. 2017

FEDERATED AVERAGING



1. Select a random set of K clients
2. Broadcast $w^{(t)}$
3. Perform E iterations of SGD locally as $w_k^{(t)} \leftarrow w_k - \eta \nabla \mathcal{L}(w; b)$
4. Send $w_k^{(t)}$ back to the server
5. Aggregate updates as $w^{(t+1)} \leftarrow \sum_{k=1}^{K} \frac{n_k}{n} w_k^{(t)}$
6. If not converged, go to 1.

H. B. McMahan *et al.*, "Communication-Efficient Learning of Deep Networks from Decentralized Data," *arXiv:1602.05629 [cs]*, Feb. 2017

OUTLINE

Federated Averaging	Introduction
Types of Federated Learning	
Federated Learning as Distributed ERM	
Statistical and System Heterogeneity	Challenges
Communication Costs	
Threat Model	Security
Privacy Preservation Techniques	

FEDERATED LEARNING DIMENSIONS

Federated Learning

- Network Topology → Centralized, Clustered, Fully-decentralized
- Data Availability → Cross-silo, Cross-device
- Data Partition → Horizontal, Vertical, Transfer Learning
- Open-Source Framework → Tensorflow Federated, PySift, FATE

V. Mothukuri *et al.*, “A survey on security and privacy of federated learning,” *Future Generation Computer Systems*, vol. 115, pp. 619–640, Feb. 2021

NETWORK TOPOLOGY

Centralized Federated Learning

- Trusted third party to monitor and manage the learning process
- All clients directly communicate to the central server
- Aggregation occurs on the server

V. Mothukuri *et al.*, “A survey on security and privacy of federated learning,” *Future Generation Computer Systems*, vol. 115, pp. 619–640, Feb. 2021

NETWORK TOPOLOGY

Centralized Federated Learning

- Trusted third party to monitor and manage the learning process
- All clients directly communicate to the central server
- Aggregation occurs on the server

Clustered Federated Learning

- Trusted third party to monitor and manage the learning process
- Clients are clustered according to their data distribution or system constraints
- Aggregation occurs on the server, but follows the clustering prescriptions

V. Mothukuri *et al.*, "A survey on security and privacy of federated learning," *Future Generation Computer Systems*, vol. 115, pp. 619–640, Feb. 2021

NETWORK TOPOLOGY

Centralized Federated Learning

- Trusted third party to monitor and manage the learning process
- All clients directly communicate to the central server
- Aggregation occurs on the server

Clustered Federated Learning

- Trusted third party to monitor and manage the learning process
- Clients are clustered according to their data distribution or system constraints
- Aggregation occurs on the server, but follows the clustering prescriptions

Fully-decentralized Federated Learning

- Peer-to-peer topology, no trusted third party
- A trusted P2P protocol substitutes the role of the central server
- Aggregation occurs on the client
- Blochckain-based update ledger

V. Mothukuri *et al.*, "A survey on security and privacy of federated learning," *Future Generation Computer Systems*, vol. 115, pp. 619–640, Feb. 2021

DATA AVAILABILITY

Distributed Machine Learning

- Data stored in a network of powerful cloud machines
- Data can be shuffled and balanced across clients
- Any client has access to any part of the dataset
- Computation is the bottleneck
- Typically, 1-1000 clients

P. Kairouz *et al.*, "Advances and Open Problems in Federated Learning," *arXiv:1912.04977 [cs, stat]*, Mar. 2021

DATA AVAILABILITY

Distributed Machine Learning

- Data stored in a network of powerful cloud machines
- Data can be shuffled and balanced across clients
- Any client has access to any part of the dataset
- Computation is the bottleneck
- Typically, 1-1000 clients

Cross-Silo Federated Learning

- Data stored in edge devices with high computational power (institutions)
- Data never leave the client
- Data can be accessed only by the owner and data samples are never explicitly shared
- Computation or communication can be the bottleneck
- Typically, 2-100 clients

P. Kairouz *et al.*, "Advances and Open Problems in Federated Learning," *arXiv:1912.04977 [cs, stat]*, Mar. 2021

DATA AVAILABILITY

Distributed Machine Learning

- Data stored in a network of powerful cloud machines
- Data can be shuffled and balanced across clients
- Any client has access to any part of the dataset
- Computation is the bottleneck
- Typically, 1-1000 clients

Cross-Silo Federated Learning

- Data stored in edge devices with high computational power (institutions)
- Data never leave the client
- Data can be accessed only by the owner and data samples are never explicitly shared
- Computation or communication can be the bottleneck
- Typically, 2-100 clients

Cross-Device Federated Learning

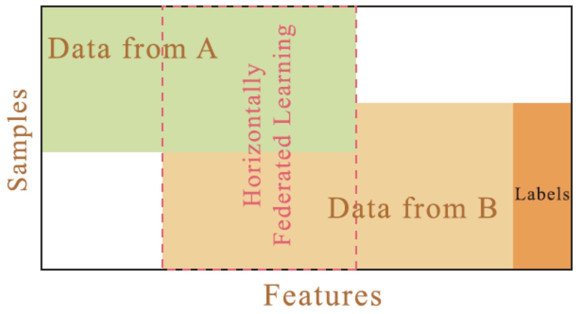
- Data stored in edge devices with low computational power (end-users)
- Data never leave the client
- Data can be accessed only by the owner and data samples are never explicitly shared
- Communication is the bottleneck
- Up to 10^6 clients

P. Kairouz *et al.*, “Advances and Open Problems in Federated Learning,” *arXiv:1912.04977 [cs, stat]*, Mar. 2021

DATA PARTITION

Horizontal Federated Learning

- Features overlap a lot
- Users overlap a little
- Example: same service provider in different regions

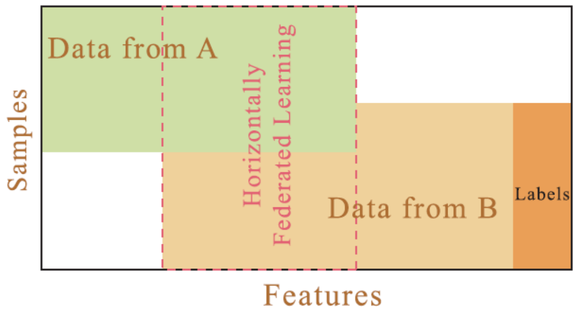


C. Zhang *et al.*, "A survey on federated learning," *Knowledge-Based Systems*, vol. 216, p. 106775, Mar. 2021

DATA PARTITION

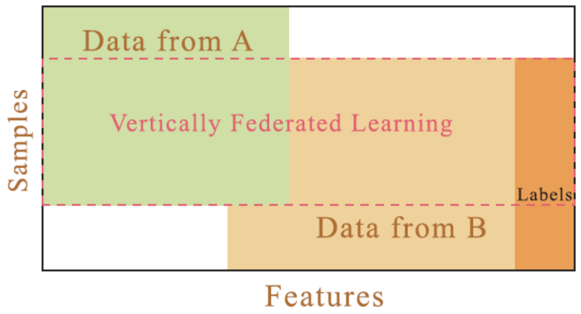
Horizontal Federated Learning

- Features overlap a lot
- Users overlap a little
- Example: same service provider in different regions



Vertical Federated Learning

- Features overlap a little
- Users overlap a lot
- Example: two different institutions, e.g., a bank and a store in the same region



C. Zhang *et al.*, "A survey on federated learning," *Knowledge-Based Systems*, vol. 216, p. 106775, Mar. 2021

DATA PARTITION

Horizontal Federated Learning

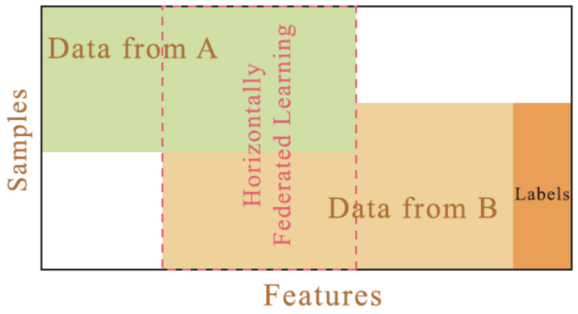
- Features overlap a lot
- Users overlap a little
- Example: same service provider in different regions

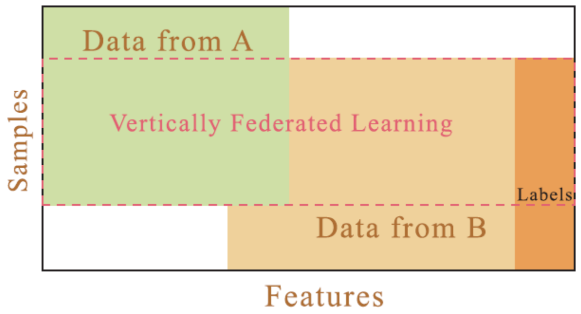
Vertical Federated Learning

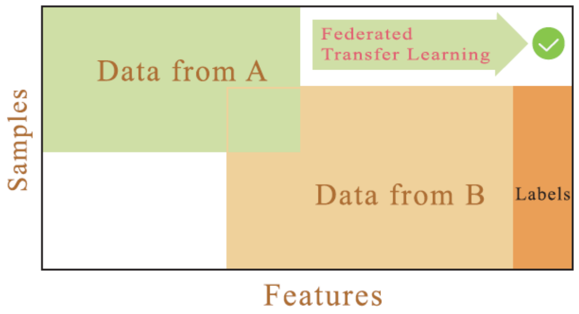
- Features overlap a little
- Users overlap a lot
- Example: two different institutions, e.g., a bank and a store in the same region

Federated Transfer Learning

- Features overlap a little
- Users overlap a little
- Example: two different institutions in different regions







C. Zhang *et al.*, "A survey on federated learning," *Knowledge-Based Systems*, vol. 216, p. 106775, Mar. 2021

OPEN-SOURCE FRAMEWORKS

Tensorflow Federated

- Integrated with LEAF
- Has a high-level API and a Federated Core for custom algorithms
- Works with Docker and Kubernetes
- Can simulate a federated network efficiently

PySift

- Based on PyTorch
- Provides a socket interface for model exchange
- Supports asynchronous FL
- Supports encryption and differential privacy

FATE

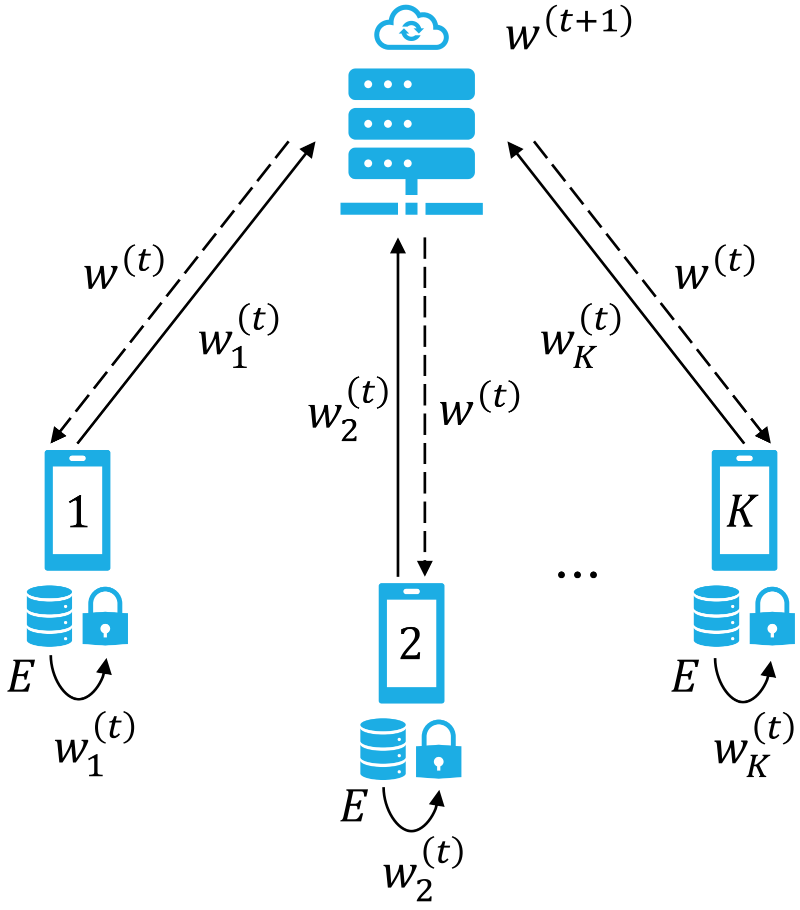
- Federated secure computing framework for distributed ML
- Supports industrial-level deployment
- Tracking FL applications with FATEBoard visualizations
- Works with Docker and Kubernetes

V. Mothukuri *et al.*, “A survey on security and privacy of federated learning,” *Future Generation Computer Systems*, vol. 115, pp. 619–640, Feb. 2021

OUTLINE

Federated Averaging	Introduction
Types of Federated Learning	
Federated Learning as Distributed ERM	
Statistical and System Heterogeneity	Challenges
Communication Costs	
Threat Model	Security
Privacy Preservation Techniques	

FEDERATED LEARNING AS DISTRIBUTED ERM

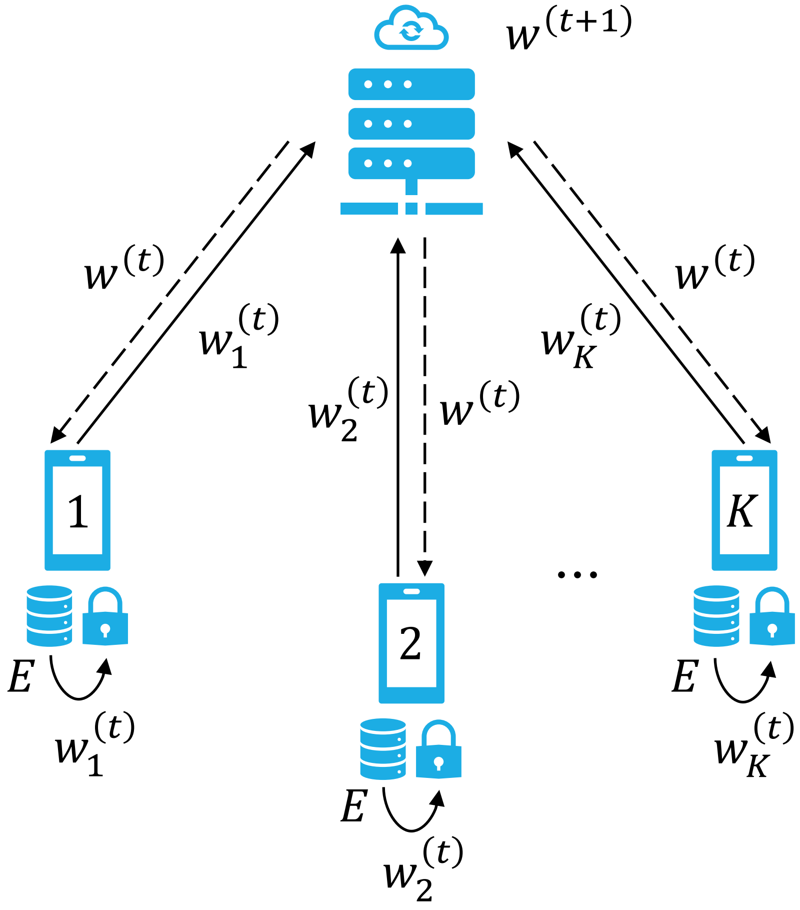


$$\min_w F(w) = \min_w \sum_{k=1}^{K} p_k\, F_k(w)$$

s.t. $\sum_{k=1}^{K} p_k = 1$ and $p_k \geq 0$.

Usually, $p_k = \frac{n_k}{\sum_{k=1}^{K} n_k} = \frac{n_k}{n}$.

FEDERATED LEARNING AS DISTRIBUTED ERM



$$\min_w F(w) = \min_w \sum_{k=1}^{K} p_k\, F_k(w)$$

In FedAvg, $F(w)$ is minimized with respect to the empirical distribution $U = \sum_{k=1}^{K} \frac{n_k}{n} D_k$

Does U really reflect the test distribution? What guarantees do we have?

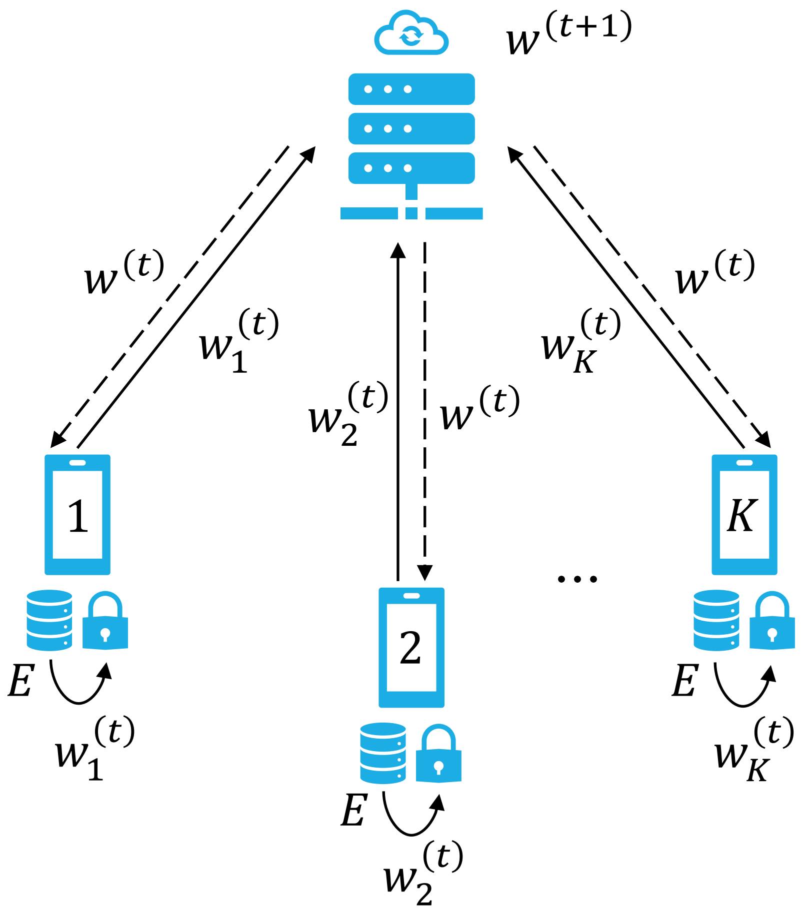
How to deal with a huge heterogeneous network of devices?

M. Mohri *et al.*, "Agnostic Federated Learning," *arXiv:1902.00146 [cs, stat]*, Jan. 2019

OUTLINE

Federated Averaging Types of Federated Learning Federated Learning as Distributed ERM	Introduction
Statistical and System Heterogeneity Communication Costs	Challenges
Threat Model Privacy Preservation Techniques	Security

STATISTICAL AND SYSTEM HETEROGENEITY



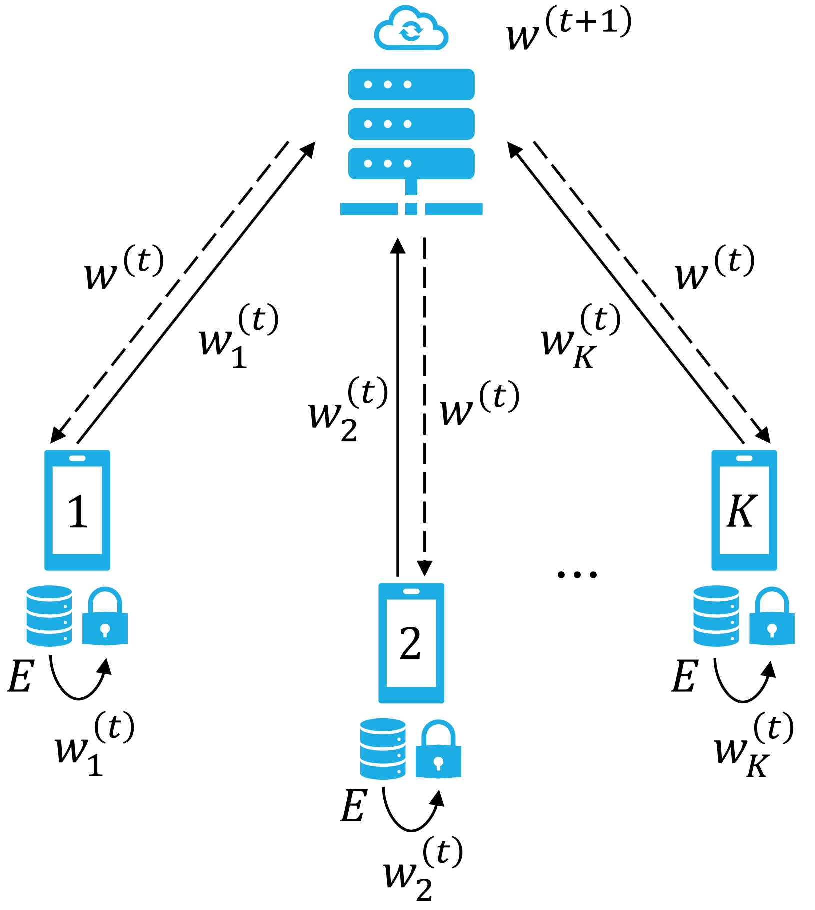
Federated Averaging

✓ Simple and easy to understand

✓ Works well in practice

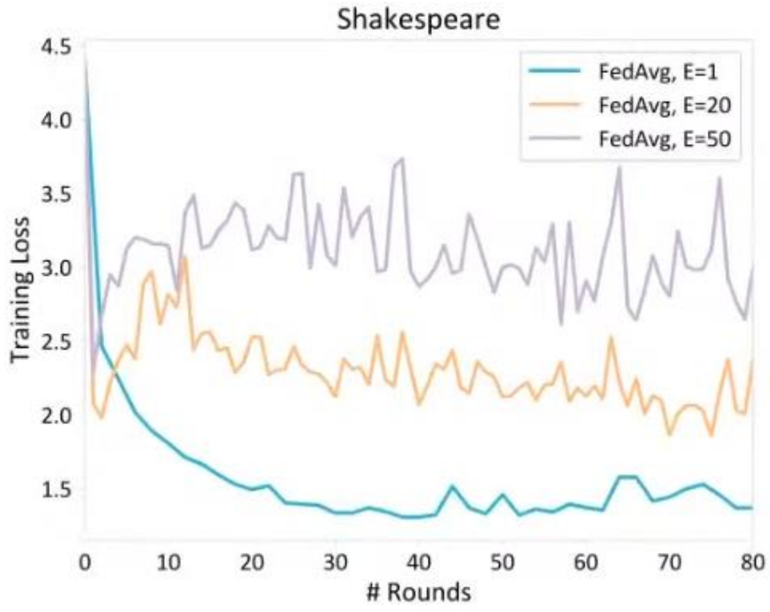
✗ Can diverge in heterogeneous settings

STATISTICAL AND SYSTEM HETEROGENEITY



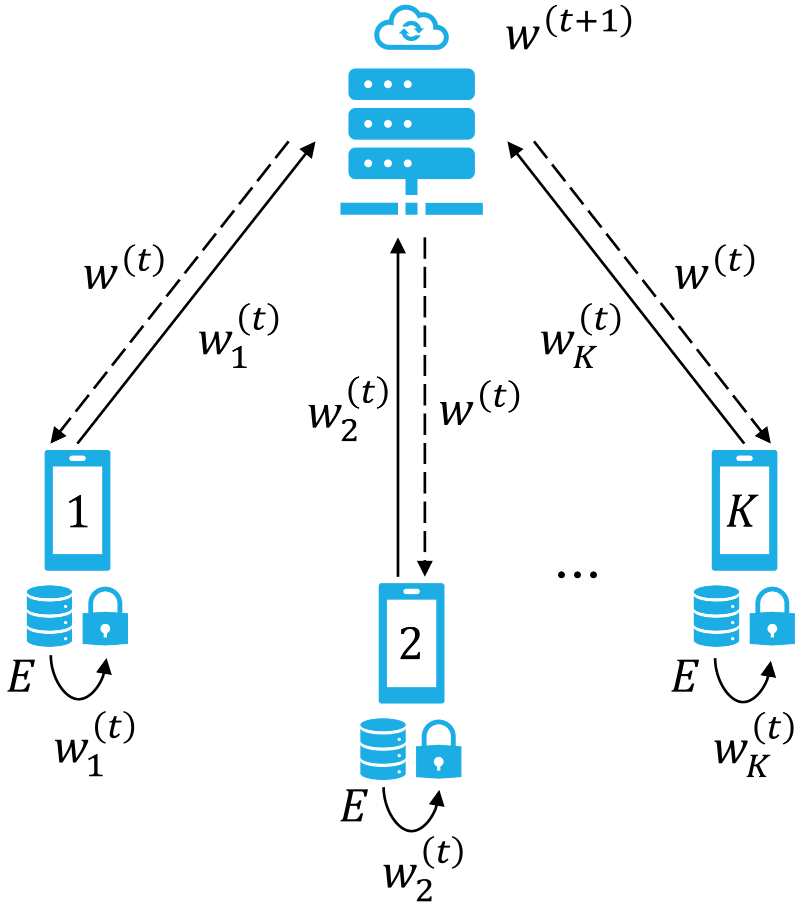
Federated Averaging

✗ Statistical heterogeneity



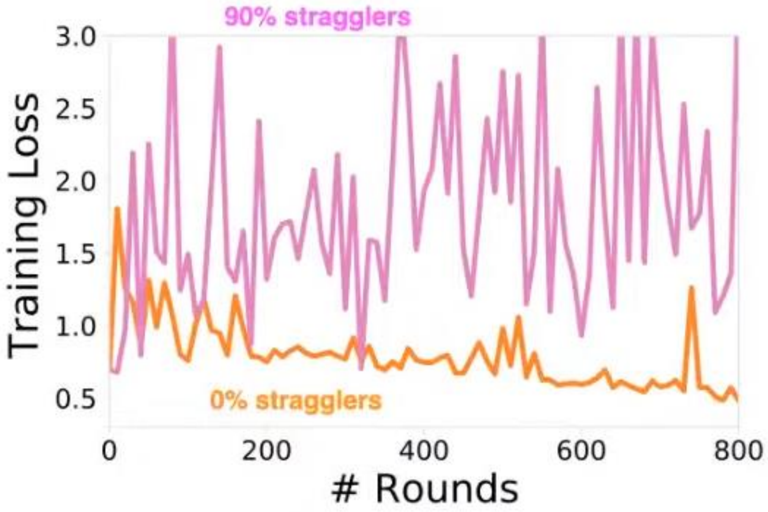
V. Smith, "On Heterogeneity in Federated Settings," *Ep. 3 of Stanford MLSys Seminar Series*, Oct. 2020

STATISTICAL AND SYSTEM HETEROGENEITY



Federated Averaging

✗ System heterogeneity



STATISTICAL AND SYSTEM HETEROGENEITY

New local update:

$$\underbrace{\min_{w} F_k(w)}_{\text{Original local objective}} + \underbrace{\frac{\mu}{2}\left\|w - w^{(t)}\right\|^2}_{\text{Regularization term to discourage big changes}}$$

FedProx

✓ Statistical heterogeneity: encourage well-behaved updates using a regularization term

✓ System heterogeneity: allow for incomplete updates after a fixed ΔT

T. Li *et al.*, "Federated Optimization in Heterogeneous Networks," *arXiv:1812.06127 [cs, stat]*, Apr. 2020

STATISTICAL AND SYSTEM HETEROGENEITY

New local update:

$$\min_w F_k(w) + \cancel{\frac{\mu}{2}\left\|w - w^{(t)}\right\|^2}$$

FedProx

✓ Statistical heterogeneity: encourage well-behaved updates using a regularization term

✓ System heterogeneity: allow for incomplete updates after a fixed ΔT

✓ Generalizes FedAvg ($\mu = 0$)

T. Li *et al.*, “Federated Optimization in Heterogeneous Networks,” *arXiv:1812.06127 [cs, stat]*, Apr. 2020

STATISTICAL AND SYSTEM HETEROGENEITY

B-dissimilarity

$$\mathbb{E}_K[\|\nabla F_k(w)\|^2] \leq B \cdot \|\nabla F(w)\|^2$$

Expected objective decrease

$$\mathbb{E}_K\left[F\left(w^{(t+1)}\right)\right] \leq F\left(w^{(t)}\right) - \rho\left\|\nabla F\left(w^{(t)}\right)\right\|^2$$

FedProx

✓ Statistical heterogeneity: encourage well-behaved updates using a regularization term

✓ System heterogeneity: allow for incomplete updates after a fixed ΔT

✓ Generalizes FedAvg

✓ Theoretical convergence guarantees; asymptotically equivalent to SGD

T. Li *et al.*, "Federated Optimization in Heterogeneous Networks," *arXiv:1812.06127 [cs, stat]*, Apr. 2020

STATISTICAL AND SYSTEM HETEROGENEITY

Data distribution simplex

$$D_\lambda = \sum_{k=1}^{K} \lambda_k \cdot D_k$$

Agnostic ERM

$$F_{D_\Lambda}(w) = \max_{\lambda \in \Lambda} \sum_{k=1}^{K} \lambda_k \cdot F_k(w)$$

Agnostic FL

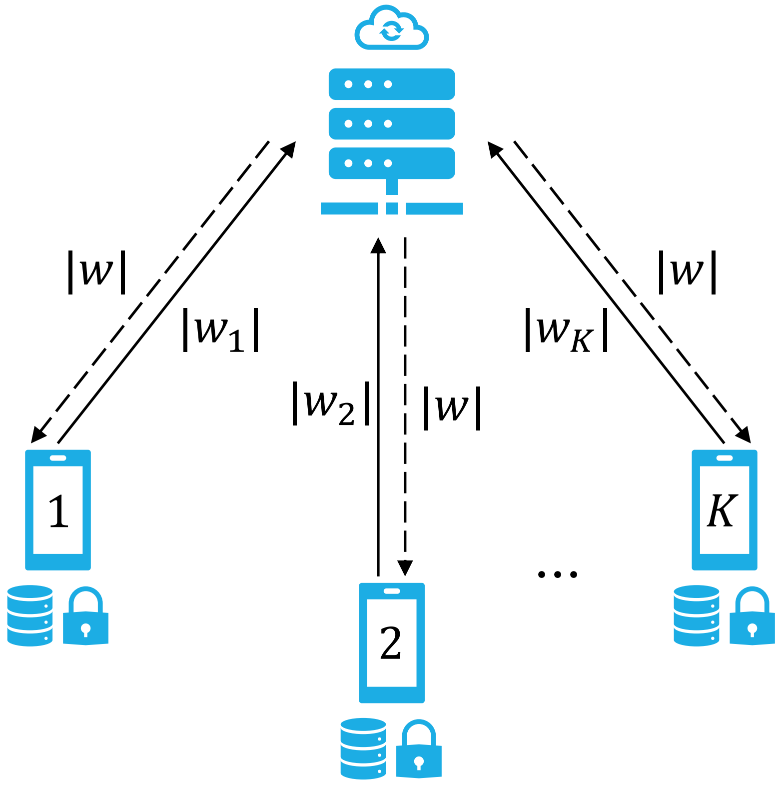
✓ Statistical heterogeneity: maximize with respect to any mixture of client distributions

✓ Fairness: under-represented clients have a role in the final model

✓ Converge bounds for convex F

OUTLINE

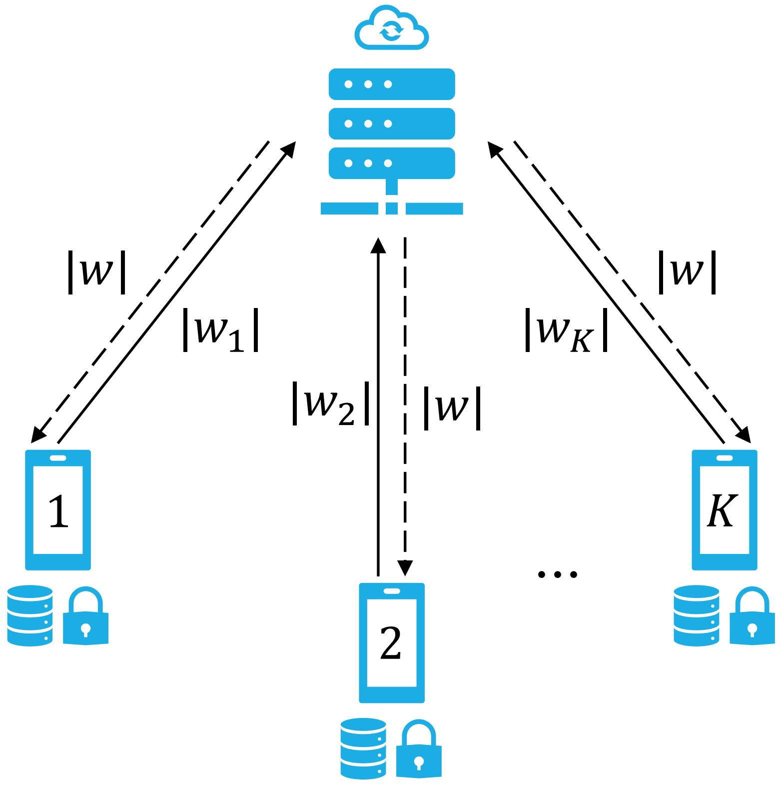
REDUCE THE COMMUNICATION COSTS



Quantization

Reduce the number of bits required for the update with discretization

REDUCE THE COMMUNICATION COSTS



Quantization

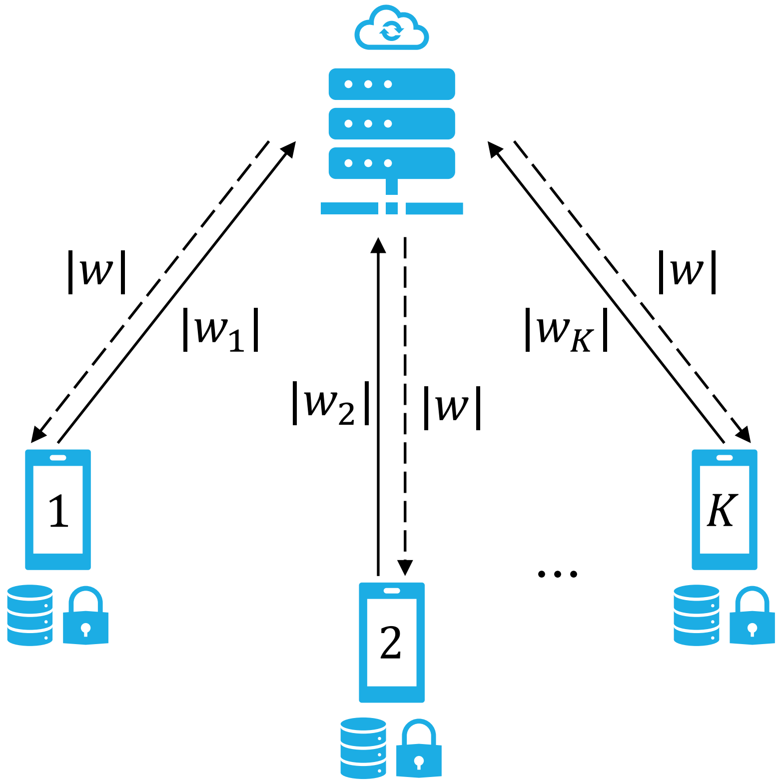
Reduce the number of bits required for the update with discretization

Less Parameters

Select and design tiny ML models to be trained in the federation

W. Y. B. Lim *et al.*, “Federated Learning in Mobile Edge Networks: A Comprehensive Survey,” *IEEE Commun. Surv. Tutorials*, vol. 22, no. 3, pp. 2031–2063, 2020

REDUCE THE COMMUNICATION COSTS



Quantization

Reduce the number of bits required for the update with discretization

Less Parameters

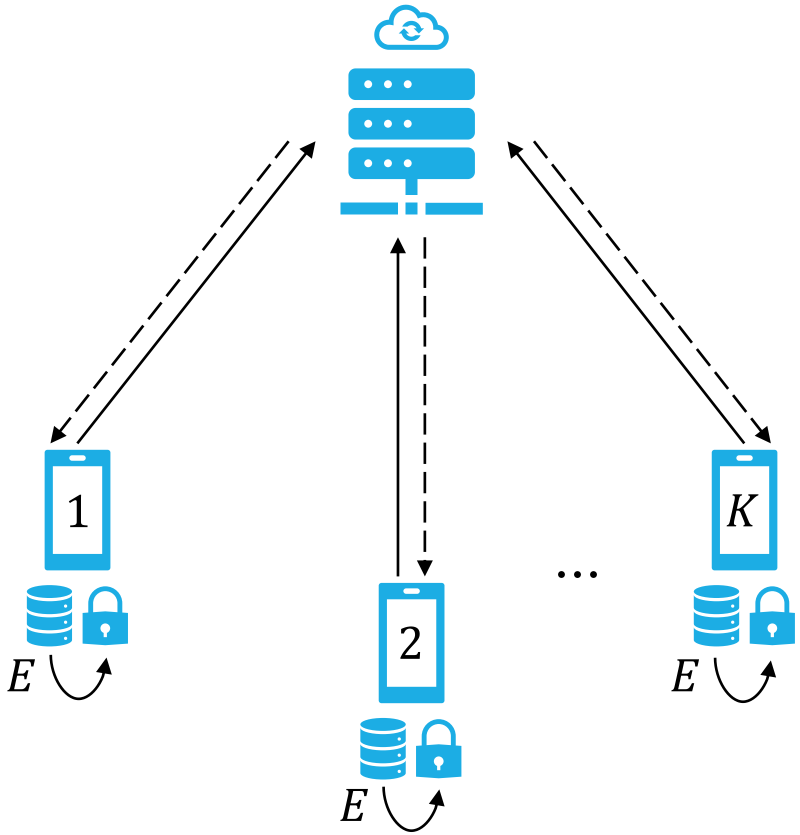
Select and design tiny ML models to be trained in the federation

Importance-based Updating

Selectively send model weights using attention-based importance metrics and dropout

W. Y. B. Lim *et al.*, "Federated Learning in Mobile Edge Networks: A Comprehensive Survey," *IEEE Commun. Surv. Tutorials*, vol. 22, no. 3, pp. 2031–2063, 2020

REDUCE THE COMMUNICATION COSTS



Increase local computation

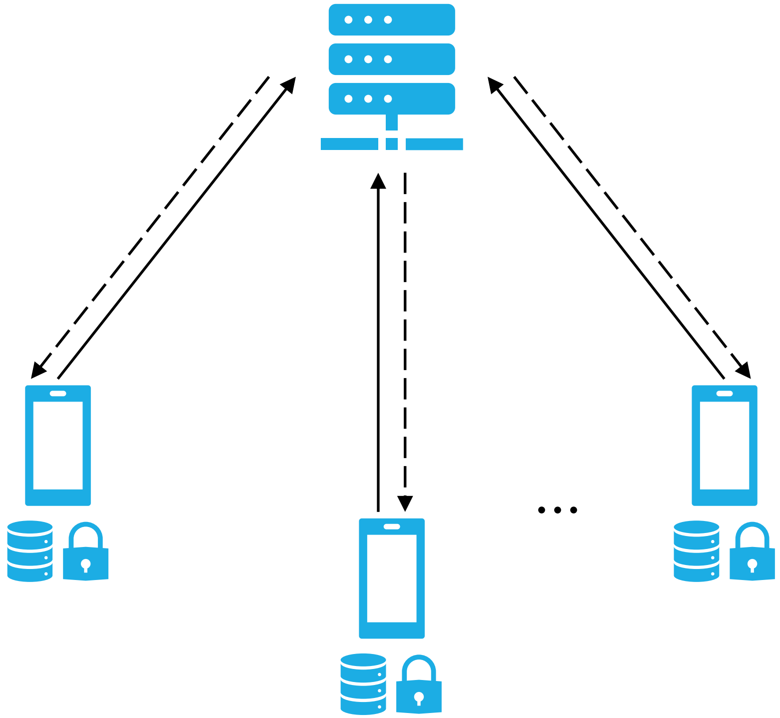
By increasing E, the learning process involves less iterations; this, however, may make convergence harder

W. Y. B. Lim *et al.*, "Federated Learning in Mobile Edge Networks: A Comprehensive Survey," *IEEE Commun. Surv. Tutorials*, vol. 22, no. 3, pp. 2031–2063, 2020

OUTLINE

Federated Averaging Types of Federated Learning Federated Learning as Distributed ERM	Introduction
Statistical and System Heterogeneity Communication Costs	Challenges
Threat Model Privacy Preservation Techniques	Security

SECURITY PILLARS OF FEDERATED LEARNING



Confidentiality

Private user data cannot be exposed

Integrity

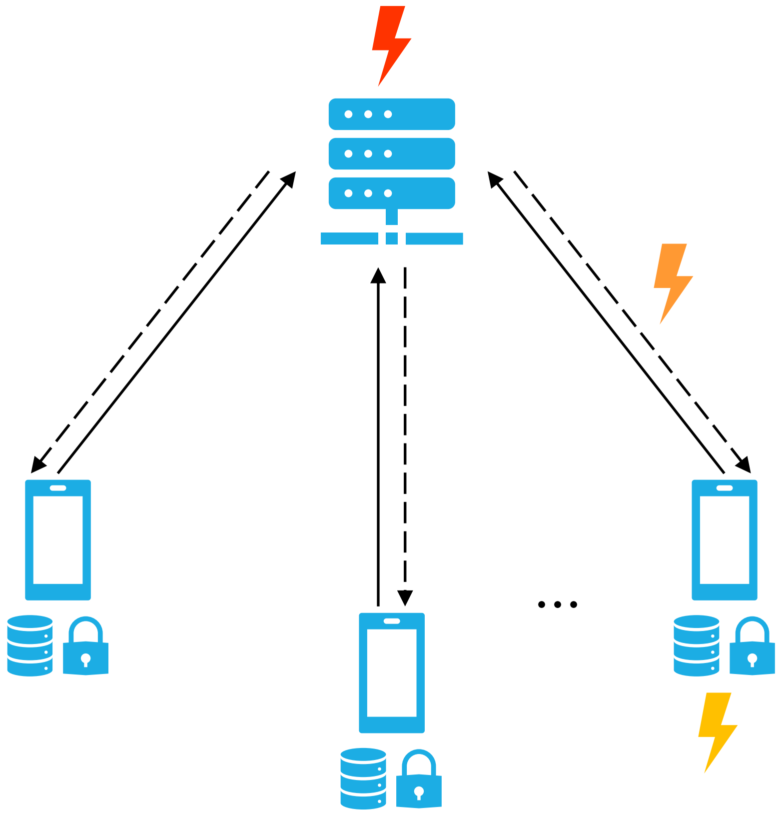
The FL algorithm must converge to the correct model, as if all participants are collaborative

Availability

The model must be accessible to all the clients in the federation

V. Mothukuri *et al.*, "A survey on security and privacy of federated learning," *Future Generation Computer Systems*, vol. 115, pp. 619–640, Feb. 2021

ATTACK SURFACES



Compromised central server

The server should be robust and secure against curious attackers

Weak aggregation algorithm

Abnormalities should be identified, and suspicious clients should be dropped

Communication protocol

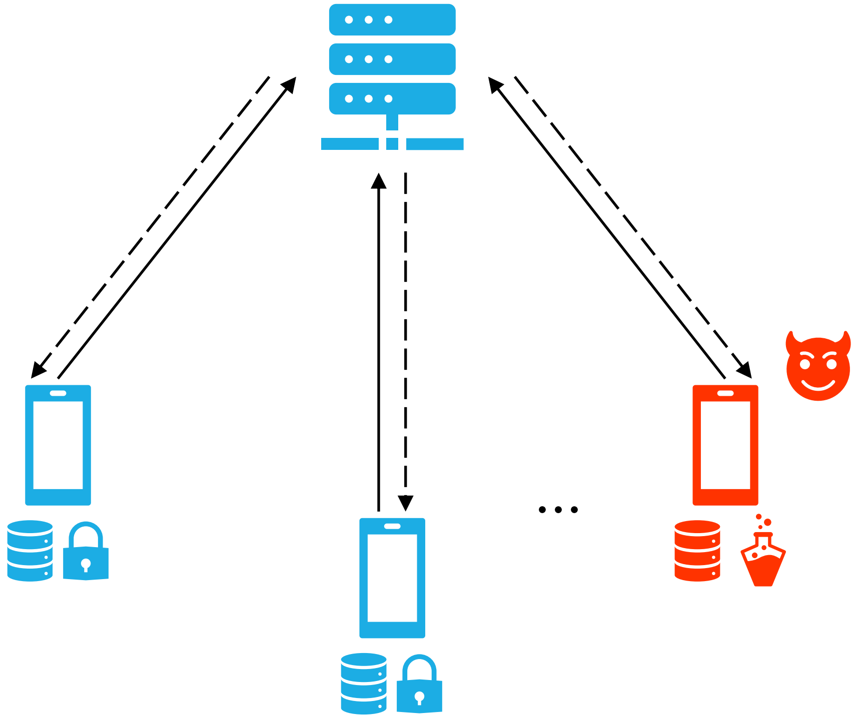
The communication channel must be reliable and secure

Client data manipulation

In large federations, clients cannot be assumed to be always honest

V. Mothukuri *et al.*, "A survey on security and privacy of federated learning," *Future Generation Computer Systems*, vol. 115, pp. 619–640, Feb. 2021

POISONING



Data Poisoning

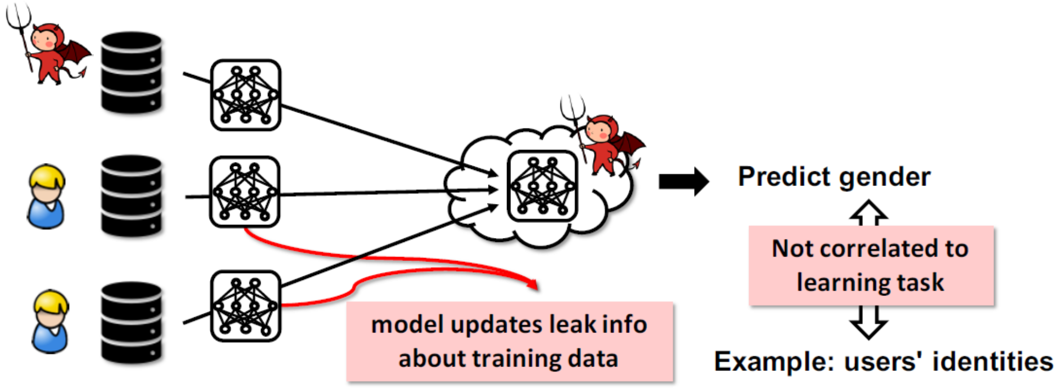
Generate dirty samples to produce falsified model parameters; a poisoned model struggles with the original task

Model Poisoning

Directly modify the model before sending it to the server; usually more effective than data poisoning, but more sophisticated

V. Mothukuri *et al.*, "A survey on security and privacy of federated learning," *Future Generation Computer Systems*, vol. 115, pp. 619–640, Feb. 2021

A MORE SUBTLE APPROACH

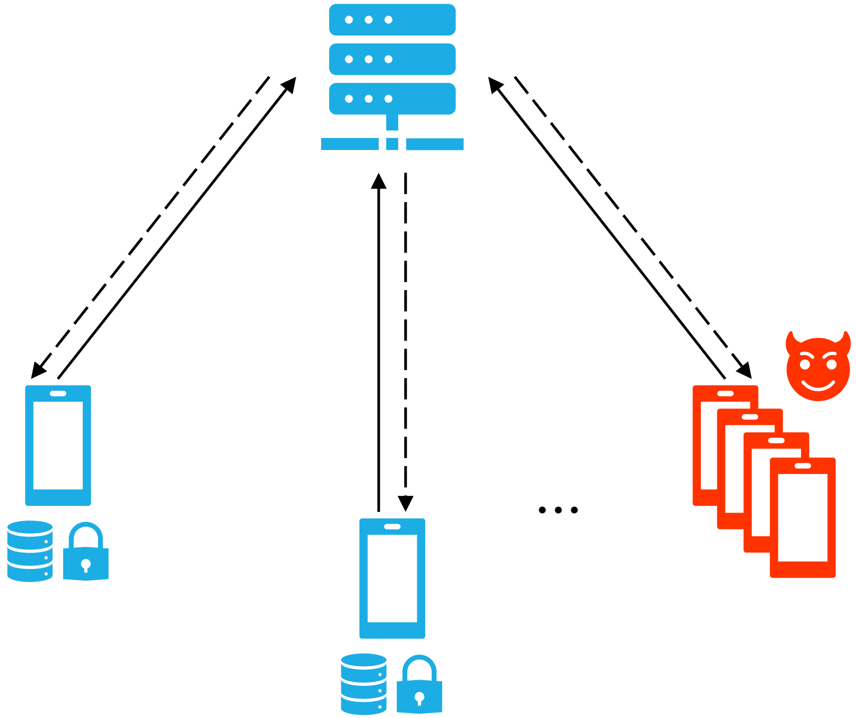


Backdoor Attack

Inject a malicious task into an existing model while retaining the overall accuracy

L. Lyu *et al.*, “Threats to Federated Learning: A Survey,” *arXiv:2003.02133 [cs, stat]*, Mar. 2020

INCREASING INFLUENCE

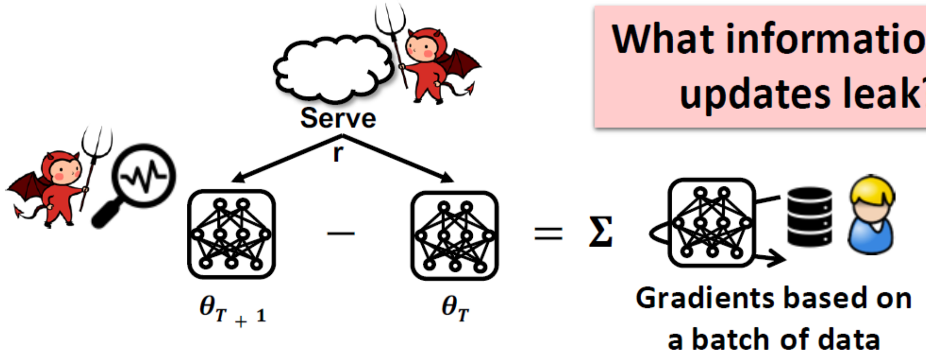


Sibyl Attack

Poisoning attack in which a malicious agent controls a swarm of dummy clients, increasing its influence in the federation

C. Fung *et al.*, “Mitigating Sybils in Federated Learning Poisoning,” *arXiv:1808.04866 [cs, stat]*, Jul. 2020

INFERENCE



What information do updates leak?

Membership Inference

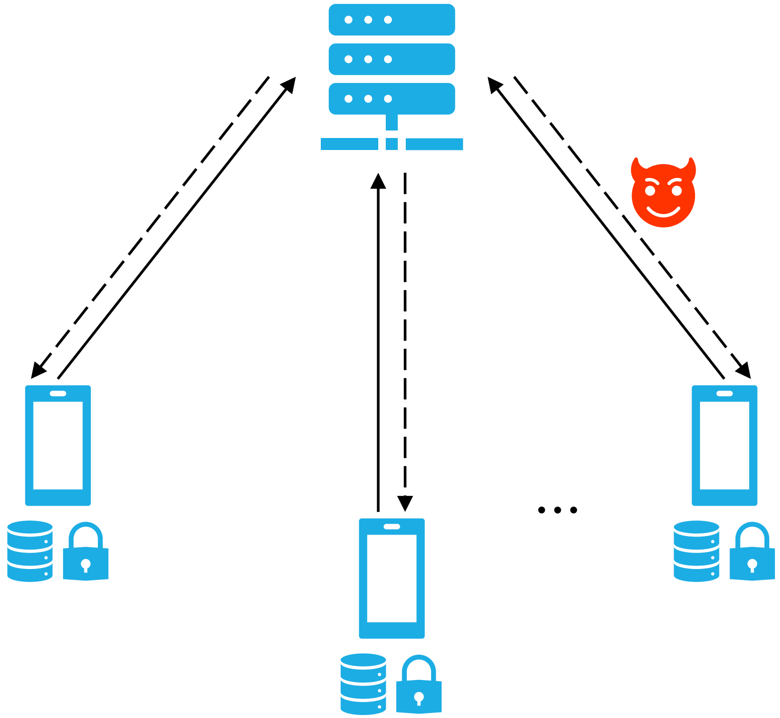
Given a data point, determine if it was used to train the model

Model Inversion

Given the output of a model, try to reconstruct which input generated it

L. Lyu *et al.*, "Threats to Federated Learning: A Survey," *arXiv:2003.02133 [cs, stat]*, Mar. 2020

COMMUNICATION BANDWIDTH



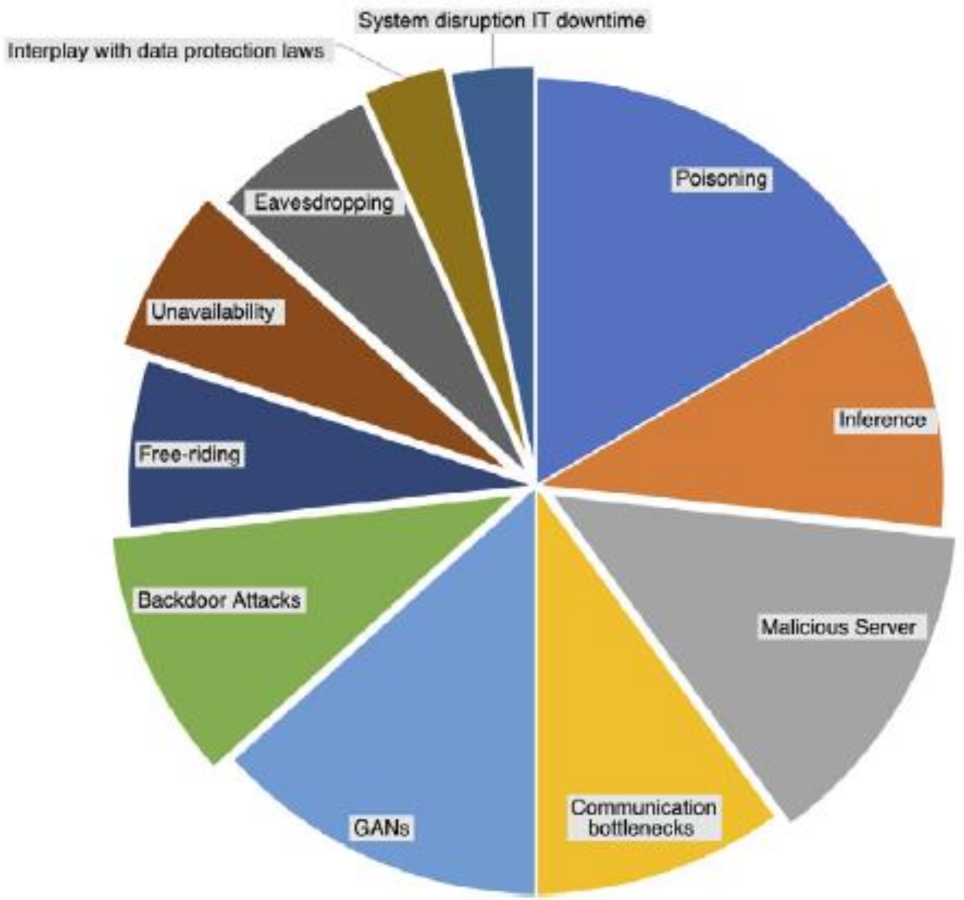
Bandwidth is the bottleneck of cross-device FL

Most FL algorithms are synchronous; therefore, malicious stragglers can disrupt the FL environment significantly

Less common, asynchronous algorithms perform well even in low-bandwidth scenarios

V. Mothukuri *et al.*, "A survey on security and privacy of federated learning," *Future Generation Computer Systems*, vol. 115, pp. 619–640, Feb. 2021

THREAT SEVERITY IN FL

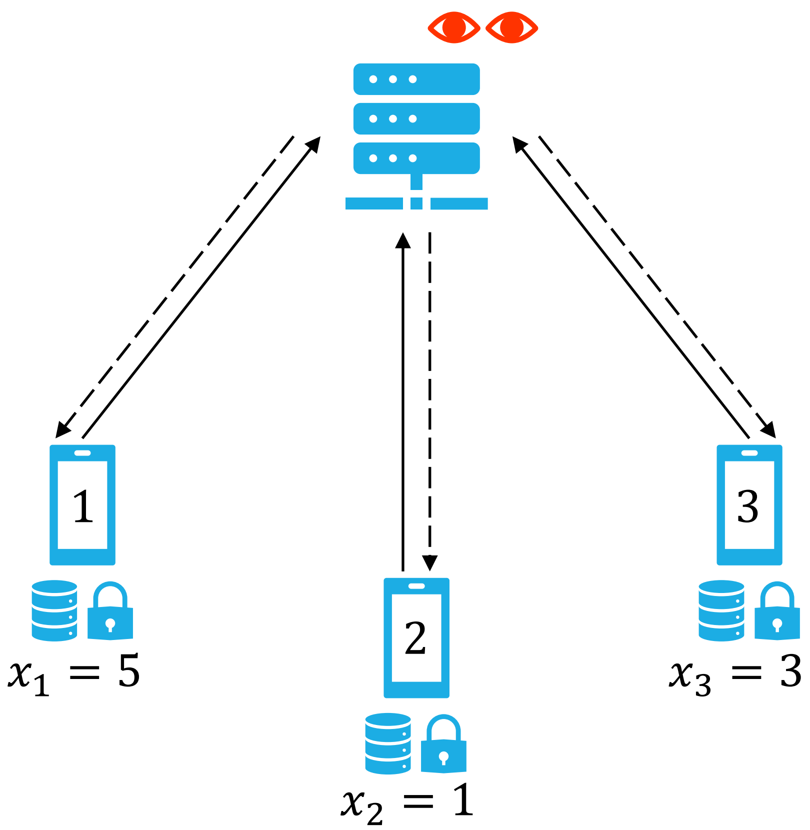


V. Mothukuri *et al.*, "A survey on security and privacy of federated learning," *Future Generation Computer Systems*, vol. 115, pp. 619–640, Feb. 2021

OUTLINE

Federated Averaging Types of Federated Learning Federated Learning as Distributed ERM	Introduction
Statistical and System Heterogeneity Communication Costs	Challenges
Threat Model Privacy Preservation Techniques	Security

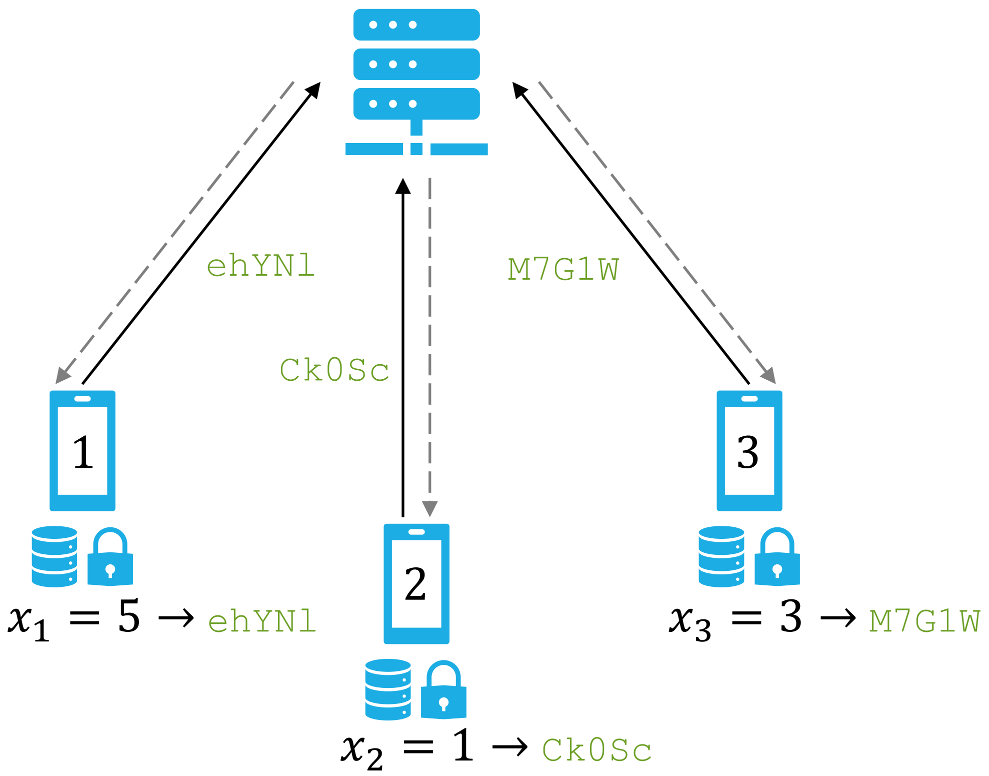
COLLABORATIVE COMPUTATION AND PRIVACY



What if the server is cooperative, but curious about our data?

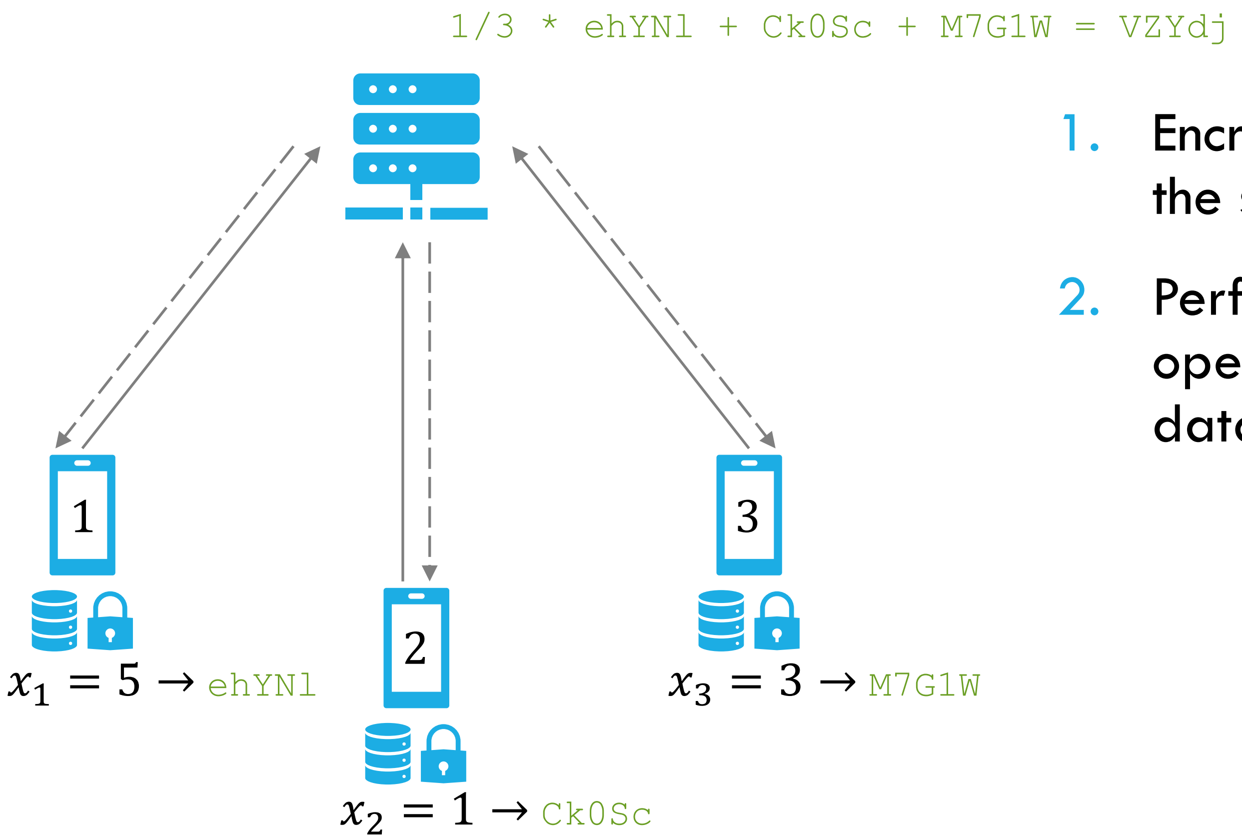
Can the server compute an aggregated function, such as $y = \frac{1}{3}\sum_{i=1}^{3} x_i$, without explicitly access any private input x_i?

HOMOMORPHIC ENCRYPTION



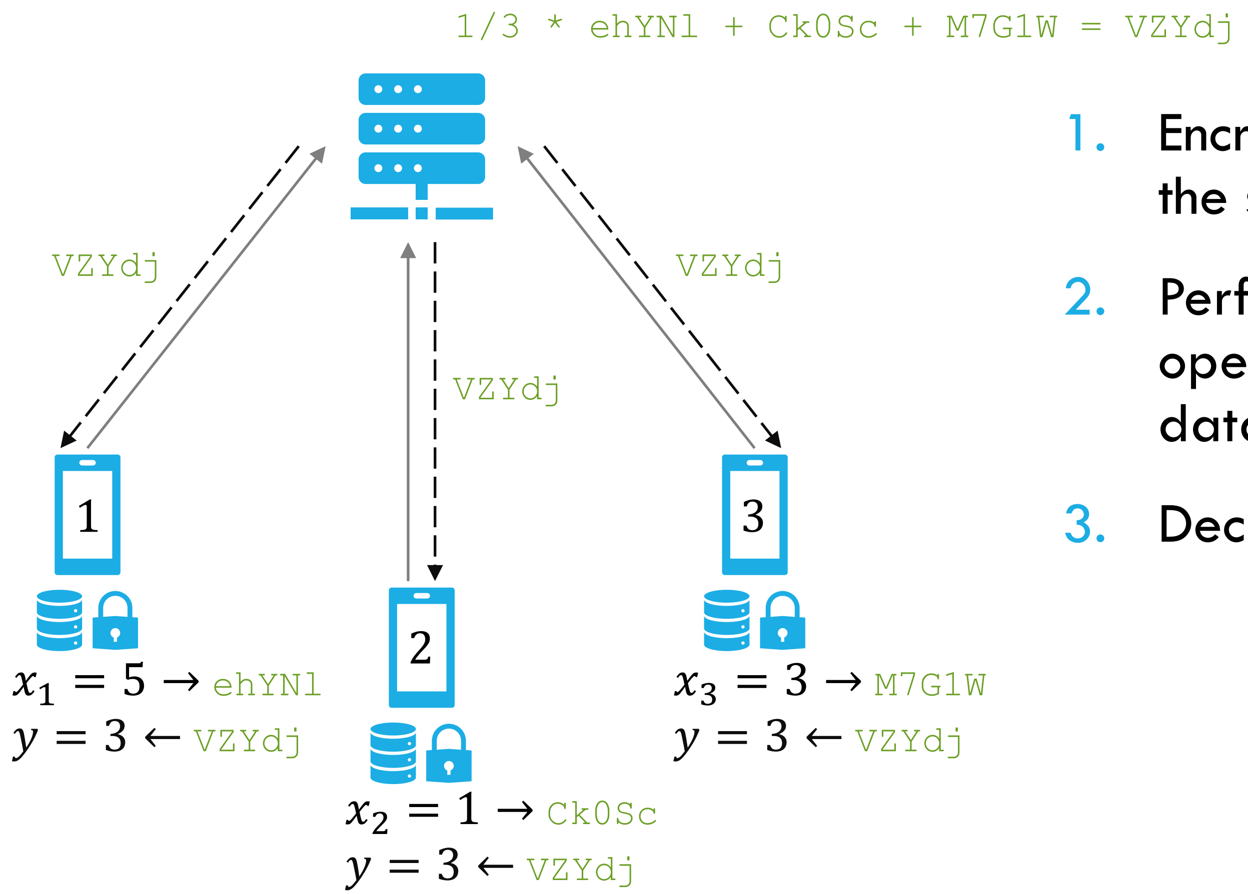
1. Encrypt private data and send it to the server

HOMOMORPHIC ENCRYPTION



1. Encrypt private data and send it to the server
2. Perform computations with custom operators directly on encrypted data

HOMOMORPHIC ENCRYPTION



1. Encrypt private data and send it to the server
2. Perform computations with custom operators directly on encrypted data
3. Decrypt the message received

SECURE MULTI-PARTY COMPUTATION

Sam: €40 **Bob**: €50 **Cassy**: €60

	Sam's data	**Bob's data**	**Cassy's data**	**Secret totals**
Sam's splits	44	-11	7	**€40**
Bob's splits				**€50**
Cassy's splits				**€60**
Shared totals				

Computes a common function on distributed data without exposing it

Example (average money):

- Sam has €40 and generates three random splits: 44, -11, and 7; he keeps 44 and sends -11 to Bob and 7 to Cassy, while keeping 44

SECURE MULTI-PARTY COMPUTATION

Sam: €40 **Bob**: €50 **Cassy**: €60

	Sam's data	**Bob's data**	**Cassy's data**	**Secret totals**
Sam's splits	44	-11	7	**€40**
Bob's splits	-6	32	24	**€50**
Cassy's splits	20	0	40	**€60**
Shared totals				

Computes a common function on distributed data without exposing it

Example (average money):

- Sam has €40 and generates three random splits: 44, -11, and 7; he keeps 44 and sends -11 to Bob and 7 to Cassy, while keeping 44
- Bob and Cassy do the same

@nawazmalla, "What is Secure Multiparty Computation?," *geeksforgeeks.org*, Jul. 2021

SECURE MULTI-PARTY COMPUTATION



Sam: €40 **Bob**: €50 **Cassy**: €60

	Sam's data	Bob's data	Cassy's data	Secret totals
Sam's splits	44	-11	7	**€40**
Bob's splits	-6	32	24	**€50**
Cassy's splits	20	0	40	**€60**
Shared totals	**€58**	**€21**	**€71**	**AVG €50**

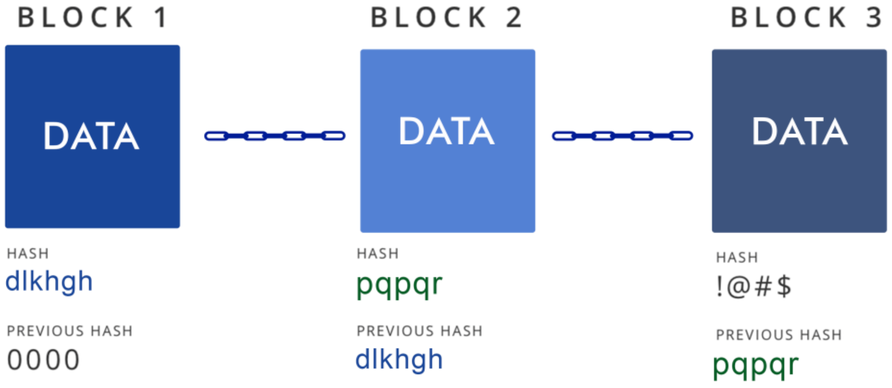
Computes a common function on distributed data without exposing it

Example (average money):

- Sam has €40 and generates three random splits: 44, -11, and 7; he keeps 44 and sends -11 to Bob and 7 to Cassy, while keeping 44
- Bob and Cassy do the same
- Everyone shares their total on shared messages and evaluates the average

@nawazmalla, "What is Secure Multiparty Computation?," *geeksforgeeks.org*, Jul. 2021

BLOCKCHAIN-BASED FEDERATED LEARNING



"Blockchain is a distributed ledger empowered by devices named miners. Each miner keeps one replica of the entire ledger locally and competes to win the opportunity to generate a new block which contains a transaction."

- ✓ No single point of failure
- ✓ Clients are authenticated
- ✓ Incentives for participation
- ✗ Computationally intensive
- ✗ Not immune to poisoning/inference

Z. Wang and Q. Hu, "Blockchain-based Federated Learning: A Comprehensive Survey," *arXiv:2110.02182 [cs]*, Oct. 2021

ADDITIONAL RESOURCES

NeurIPS 2020 Federated Learning Tutorial

https://sites.google.com/view/fl-tutorial/

Stanford MLSys Seminar (Ep. 3)

https://www.youtube.com/watch?v=laCyJICLyWg

Tensorflow Federated Tutorial Session

https://www.youtube.com/watch?v=JBNas6Yd30A

Federated Learning: An Online Comic from Google AI

https://federated.withgoogle.com/