第七章：新型分布式机器学习系统

2021年9月

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

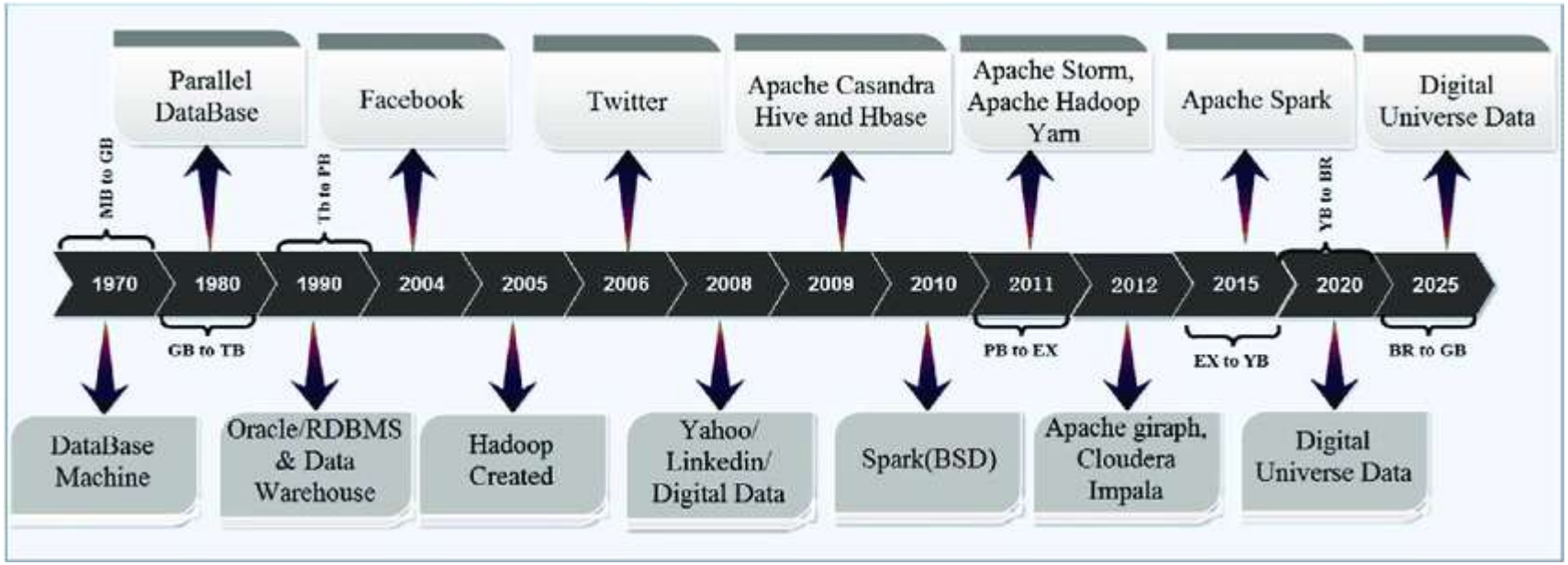
Contents

Distributed machine learning

- Big data background
- Motivation
- Architecture
- Deployment
- Shortcomings

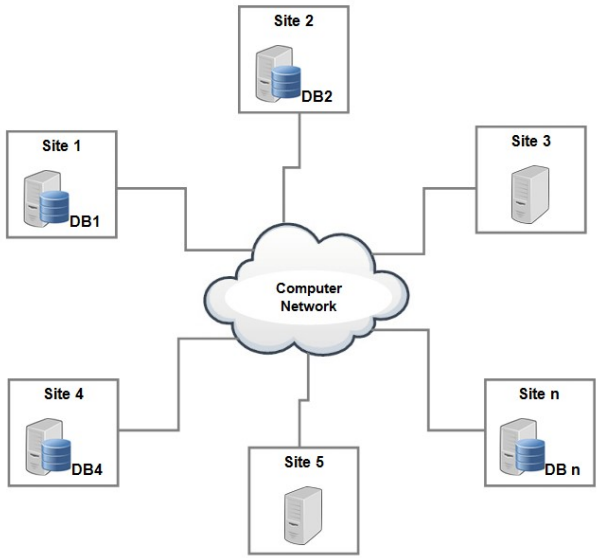
Big data background

- Since 1970, the amount of data grows larger and larger.
- To store the big data and process it, technologies such as Hadoop and Spark arisen.

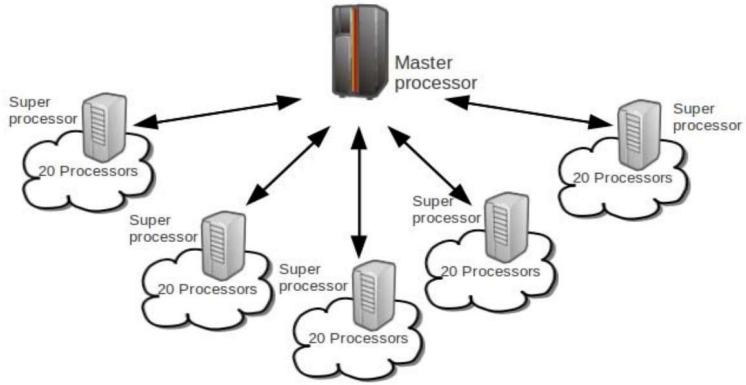


Big data background

- Because of the huge amount of data, it is not able to be stored on one machine. Distributed data storage was born.



- To process these data, distributed computing methods were utilized with the huge database.

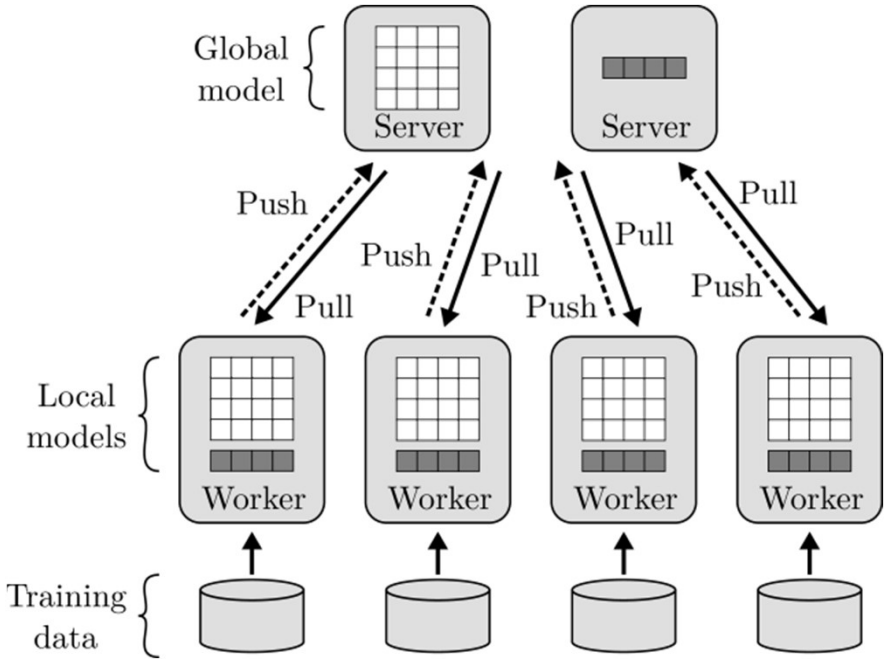


Motivation

- Machine learning is one way to extract the useful information from the data.
- However, stand-alone machine learning is not capable of the growing data.
- Distributed computing is combined with machine learning and the distributed machine learning comes.
  - Using different learning processes to train several classifiers from distributed data sets increases the possibility of achieving higher accuracy especially on a large-size domain
  - Learning in a distributed manner provides a natural solution for large-scale learning
  - inherently scalable since the growing amount of data may be offset by increasing the number of computers or processors
  - overcomes the problems of centralized storage

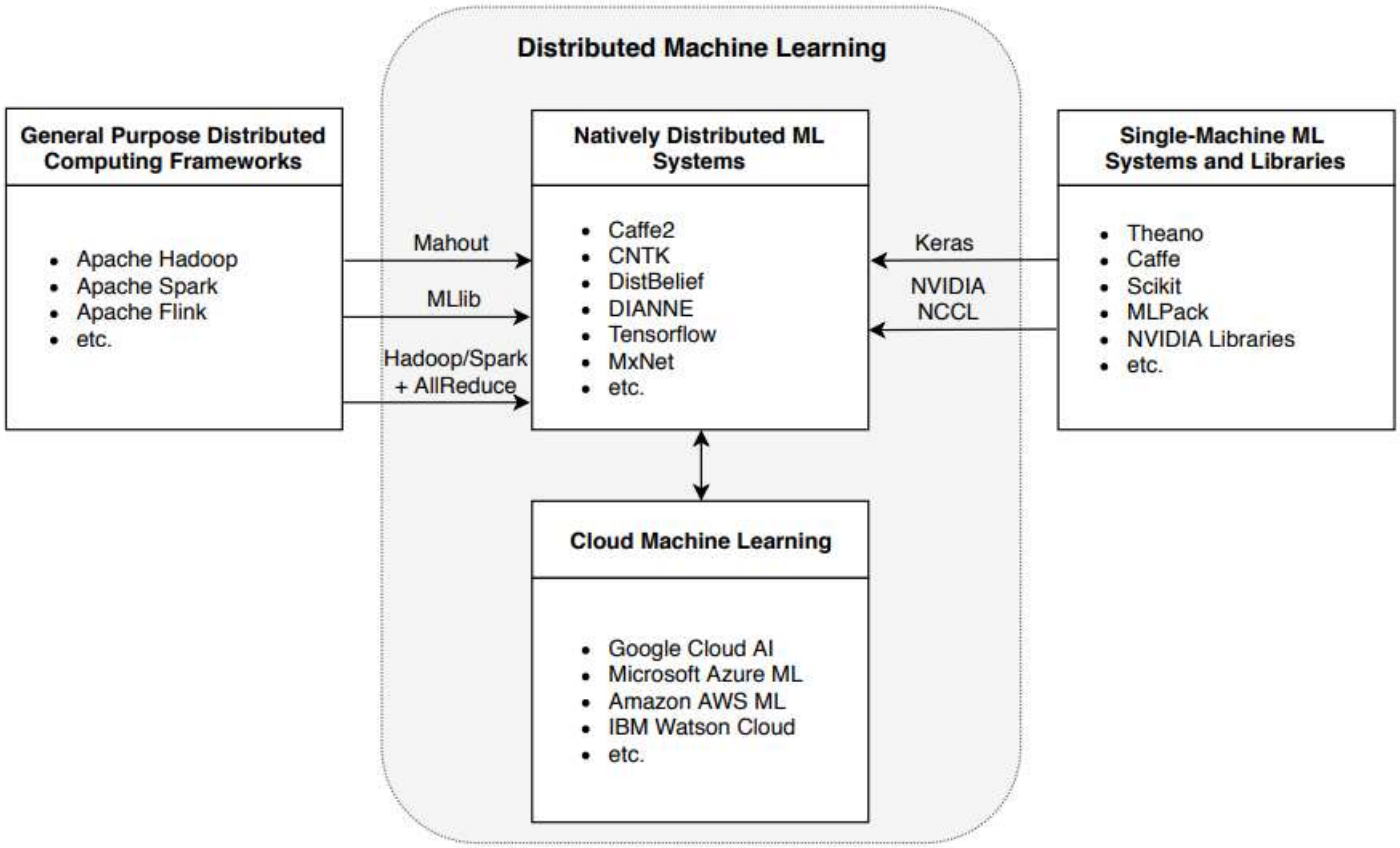
Architecture

- A server (or servers) to split data, separate it and aggregate the global model.
- Many workers with separated data to train the separated models.
- High speed networks.
- A training scheme.



Deployment

- Overview



Deployment

- MapReduce and Hadoop
  - MapReduce is a framework for processing data and was developed by Google in order to process data in a distributed setting.
  - First, all data is split into tuples during the map phase, which is followed by the reduce phase, where these tuples are grouped to generate a single output value per key.
  - MapReduce and Hadoop heavily rely on the distributed file system in every phase of the execution.
- Apache Spark
  - Apache Spark has been developed to resolve the weakness of MapReduce in transformations in linear algebra.
  - The key difference here is the MapReduce tasks, spark can keep all the data in memory, which saves expensive reads from the disk.

Deployment

- Baidu AllReduce
  - uses common high-performance computing technology to iteratively train stochastic gradient descent models on separate mini-batches of the training data.
  - linear speedup when applying this technique in order to train deep learning networks.
- Horovod
  - uses the NVIDIA Collective Communications Library (NCCL) for increased efficiency when training on (Nvidia) GPUs.
- Tensorflow
  - contains the concepts of the computation graph and parameter server.
- Caffe2
  - This deep learning framework distributes machine learning through AllReduce algorithms using NCCL.

Shortcomings

- Stable network requirement
  - Distributed machine learning **requires stable network** for model transmission both in uploads and downloads.
  - The bandwidth and speed of all the connections are assumed to be similar.
- Workers with similar abilities
  - During local training, the computation power of all the workers are better to be similar for **synchronously aggregation** of global model.
- Data splitting
  - Data is off-line stored and **outdated**.
  - The data for local training is split by the server which may cause **privacy leakage**.

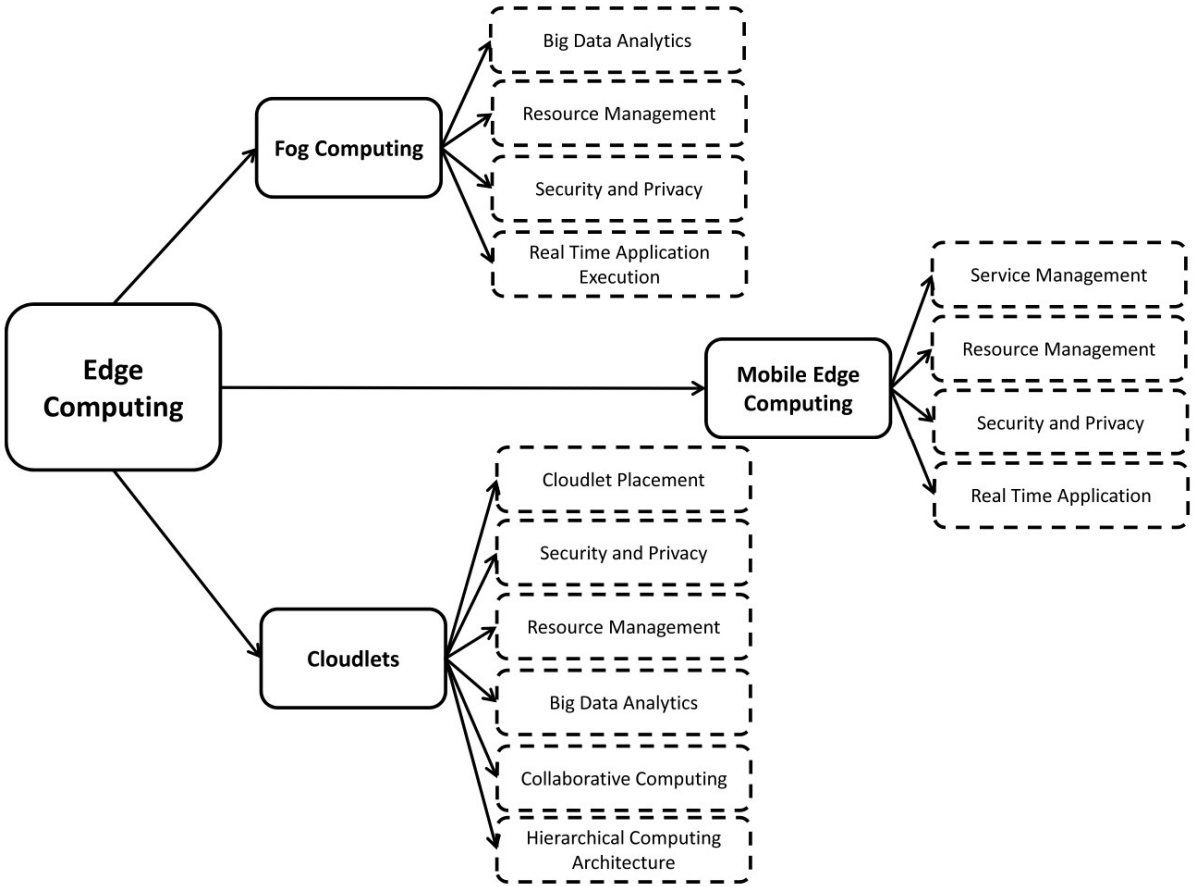
- All these features are not suitable for the era of **mobile internet** nowadays.

Edge computing

- Classification
- Features
- Architecture
- Implementations

Classification

- Overview



Features

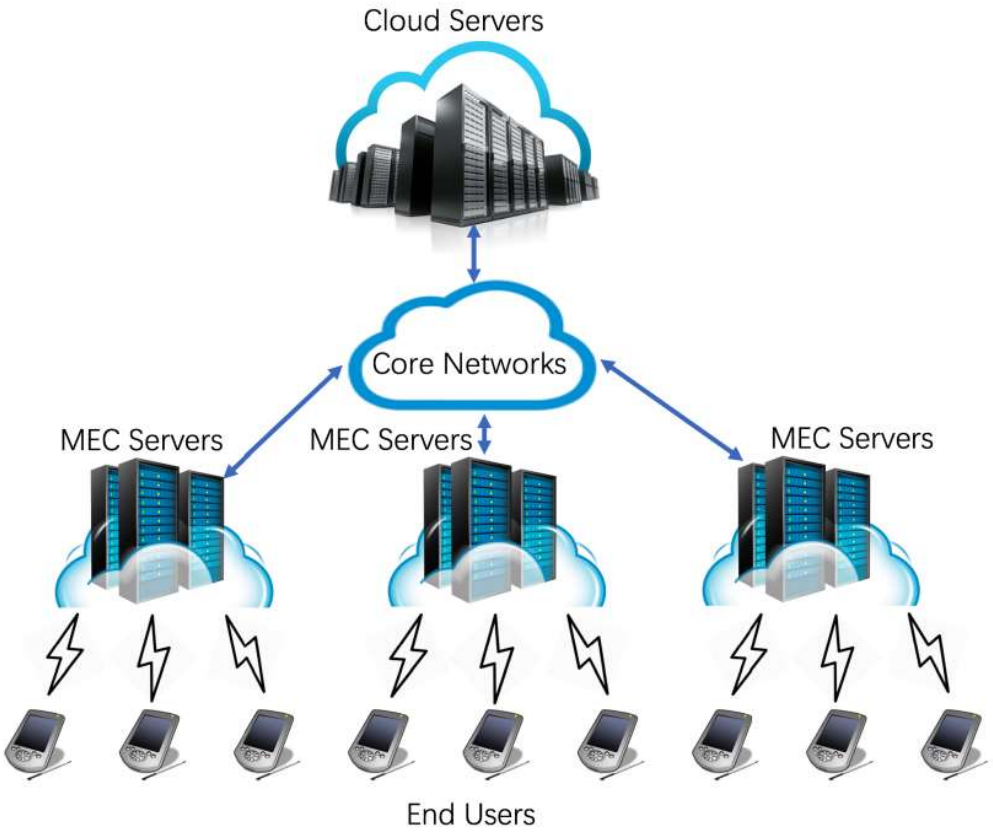
- Dense geographical distribution
  - Edge computing brings the Cloud services closer to the user by deploying numerous computing platforms in the edge networks.
- Mobility support
  - The decoupling of the host identity from the location identity constitutes the key principle that enables the mobility support in Edge computing.
- Location awareness
  - Users can employ various technologies such as cell phone infrastructure, GPS, or wireless access points to find the location of electronic devices.
- Proximity
  - The availability of the computational resources and services in the local vicinity allows the users to leverage the network context information for making offloading decisions and service usage decisions. Similar as the service provider.

Features

- Low latency
  - The low latency of Edge computing enables the users to execute their resource-intensive and delay-sensitive applications on the resource-rich Edge devices (e.g. router, access point, base station, or dedicated server).
- Context-awareness
  - Context-awareness is the characteristic of mobile devices and can be defined interdependently to location awareness.
  - The real-time network information, such as network load and user location, can be used to offer the context-aware services to the Edge users.
- Heterogeneity
  - Heterogeneity in Edge computing refers to the existence of varied platforms, architectures, infrastructures, computing, and communication technologies used by the Edge computing elements (end devices, Edge servers, and networks).
  - Edge server side-heterogeneity is mainly due to APIs, custom-built policies, and platforms.
  - The network heterogeneity refers to the diversity of communication technologies that impact the Edge service delivery.

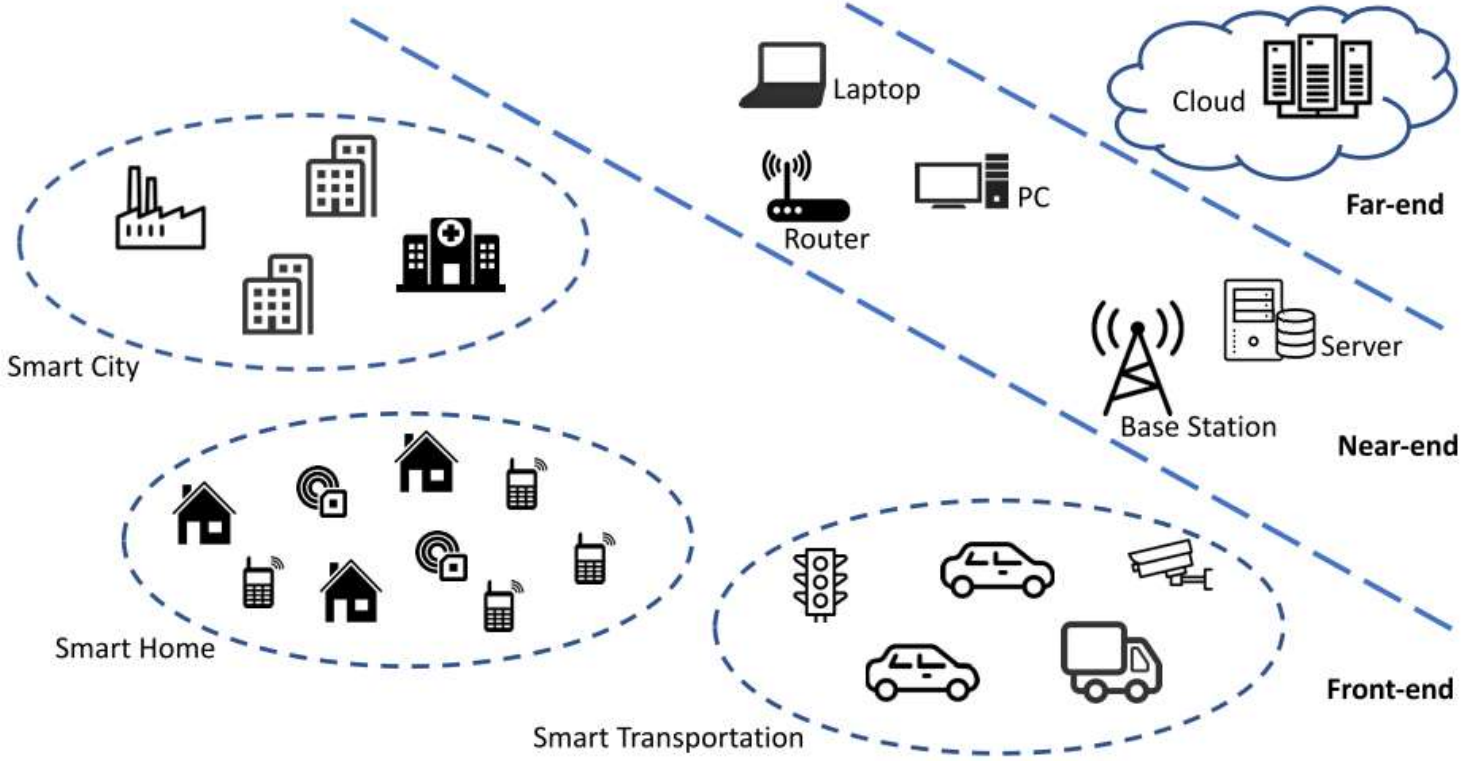
Architecture

- The basic edge computing architecture

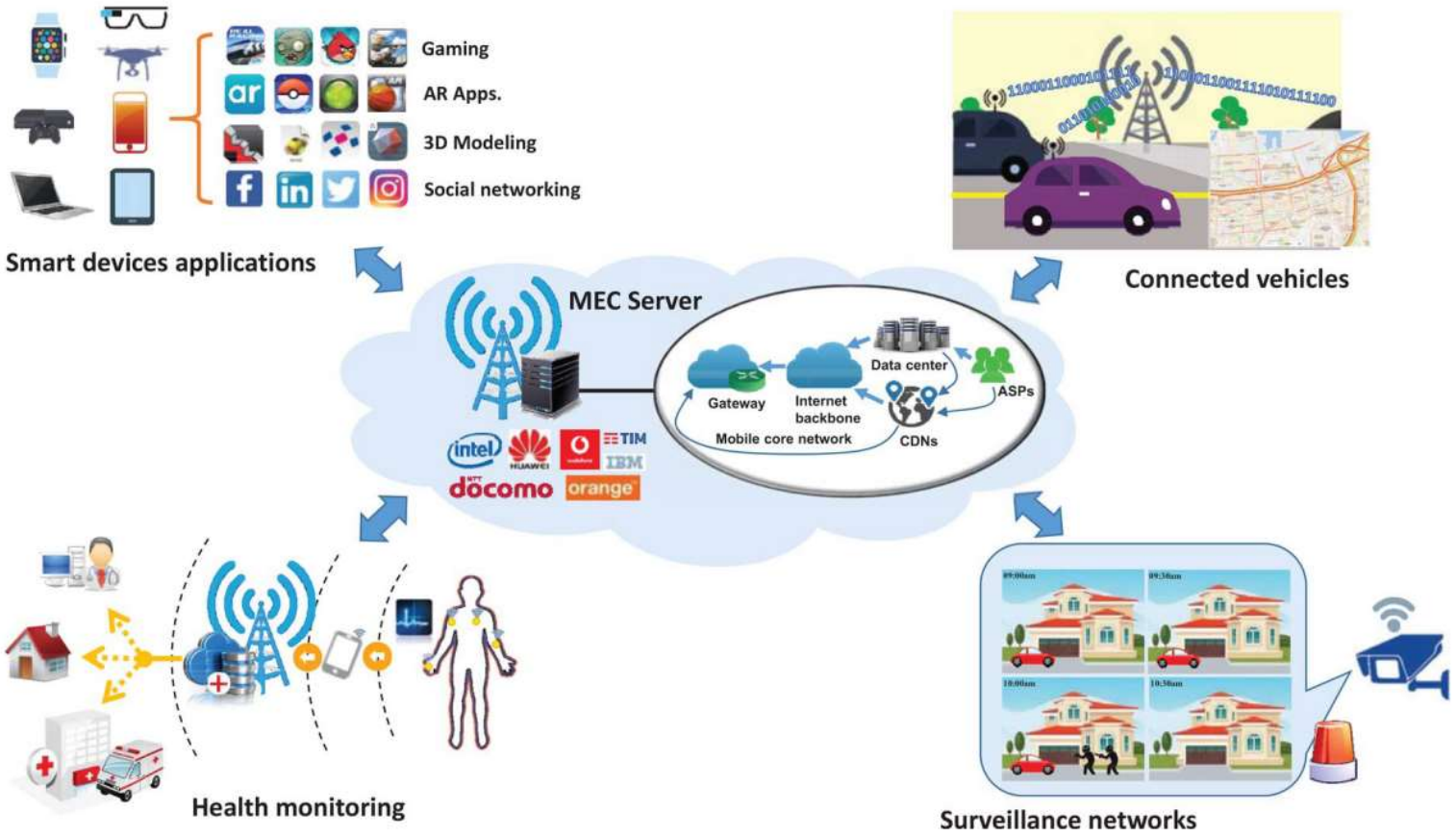


Architecture

- A typical architecture of edge computing networks



Implementations

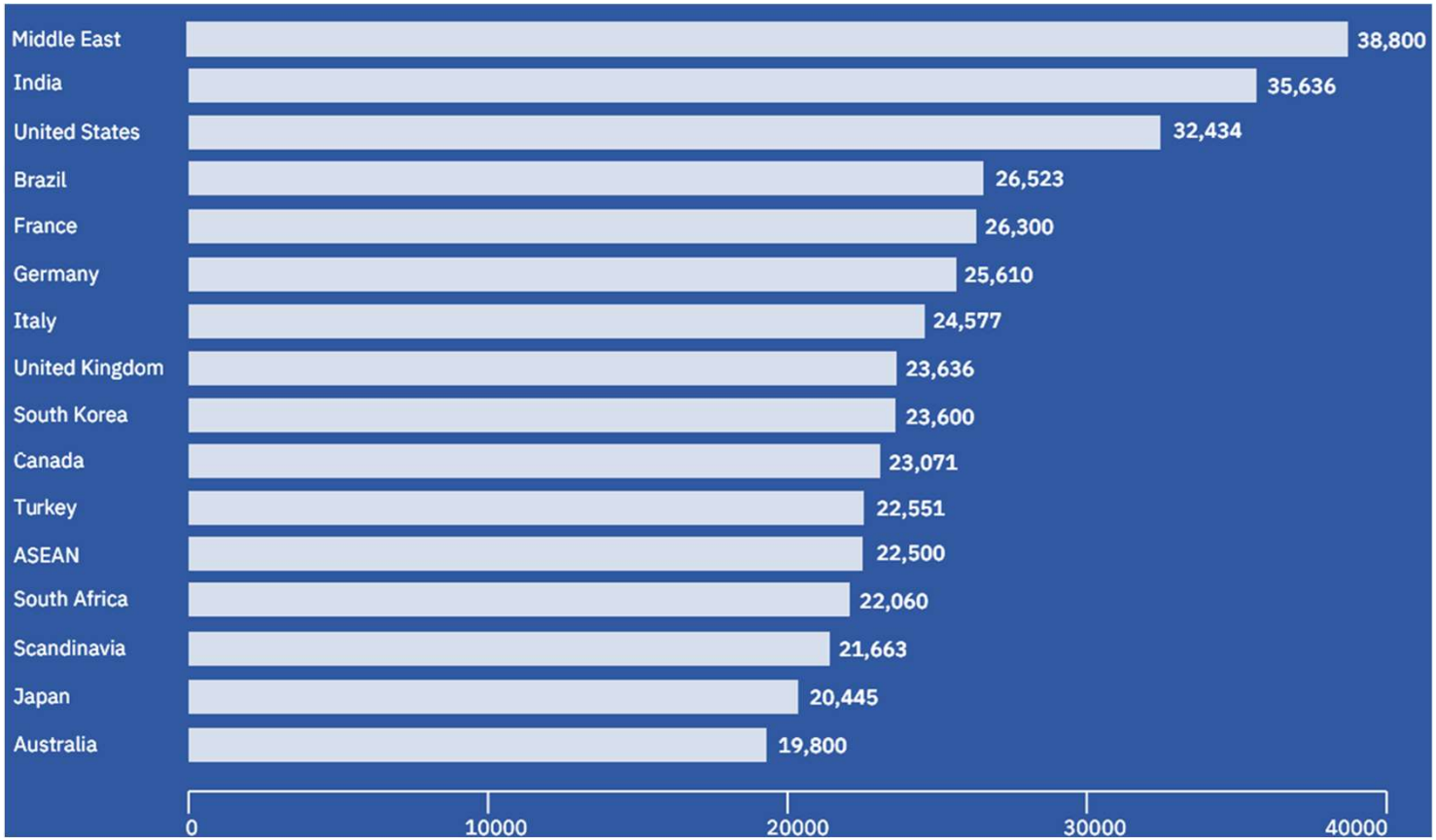


Federated learning

- Motivation
- Classification
- Architecture
- Workflow
- Challenges and opportunities

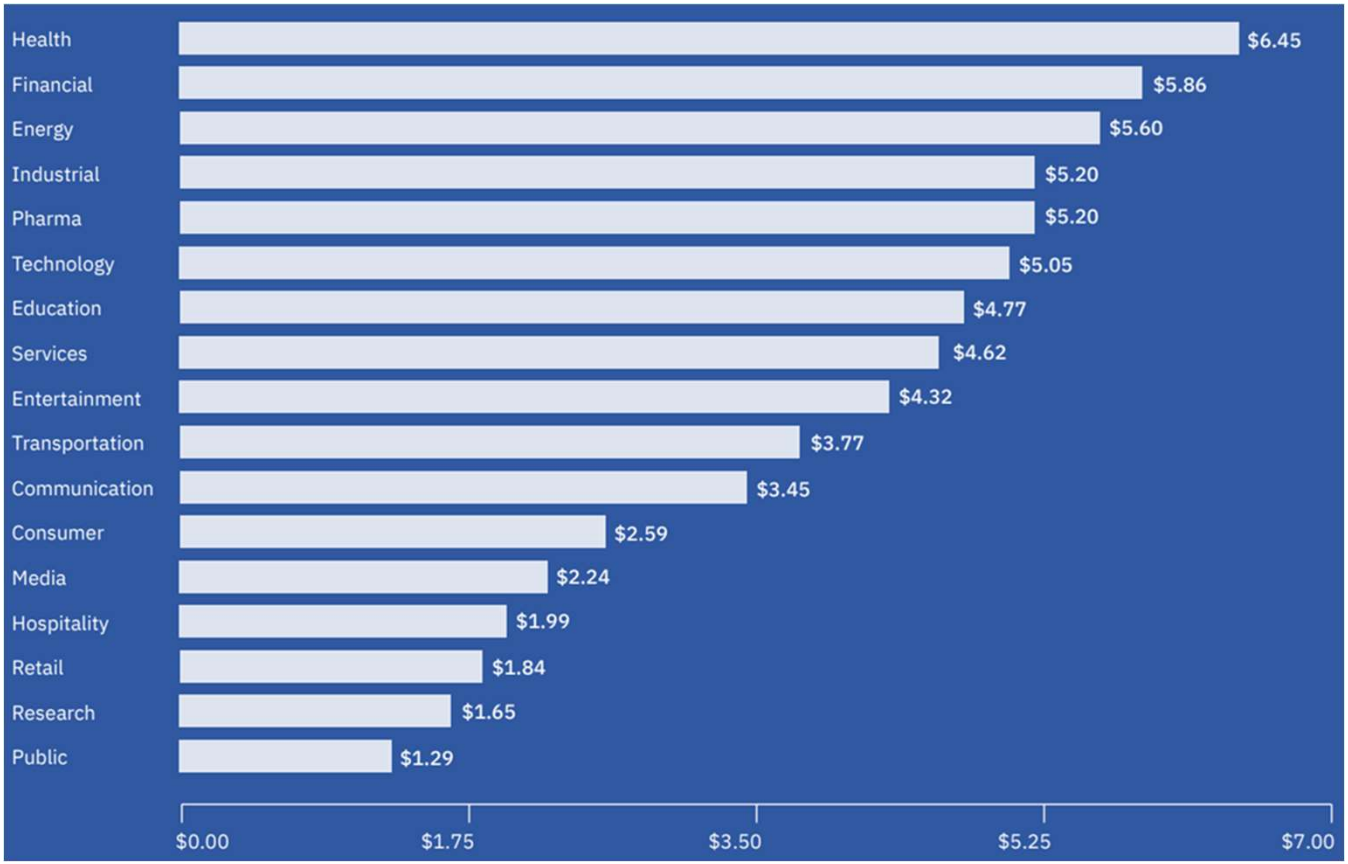
Motivation

- Data breach (quantity)



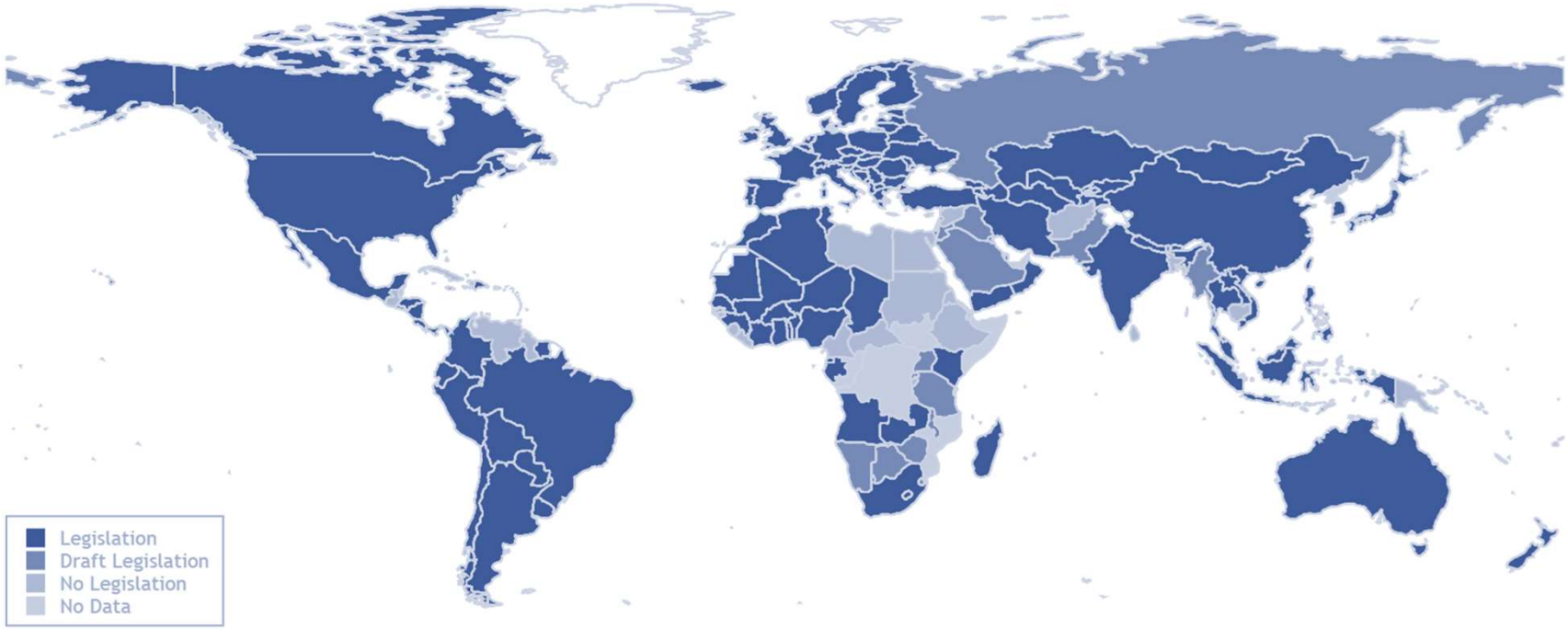
Motivation

- Data breach (cost)



Motivation

- Privacy protecting laws

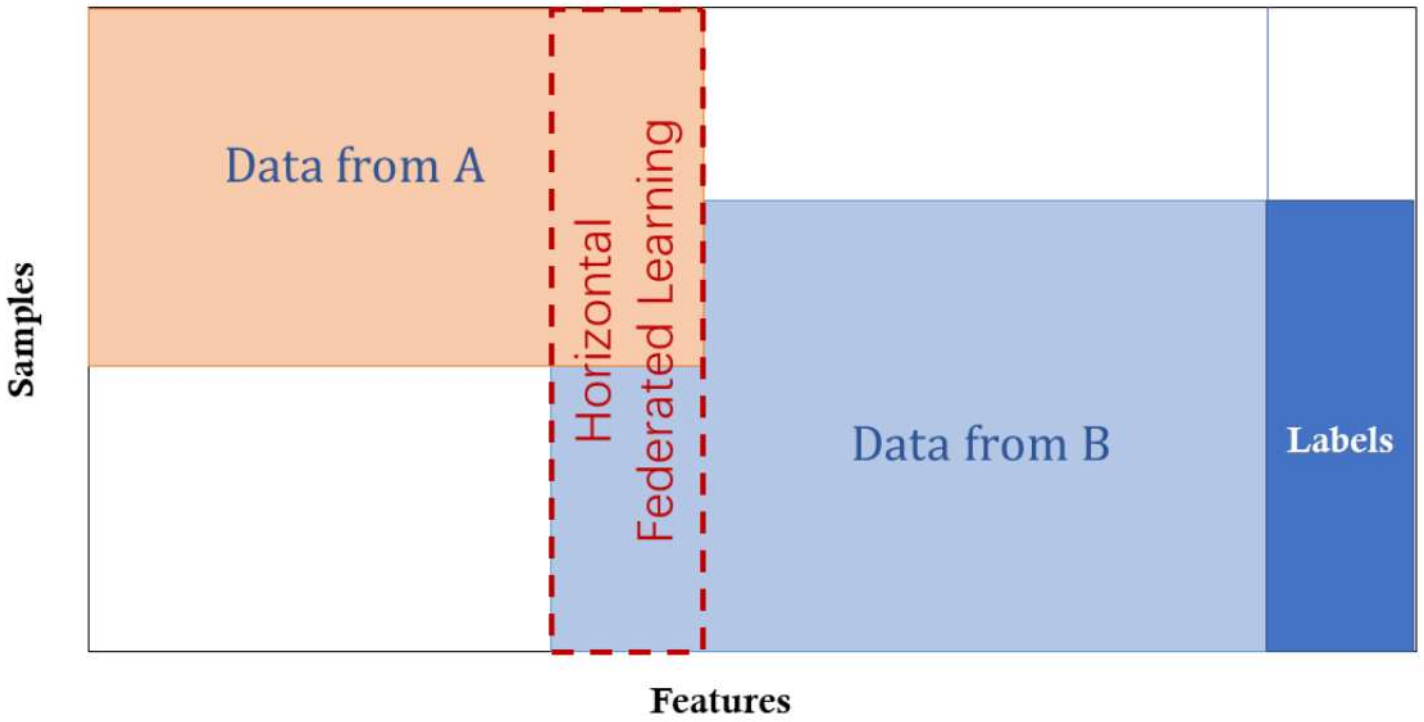


Classification

- Horizontal federated learning
- Vertical federated learning
- Federated Transfer Learning

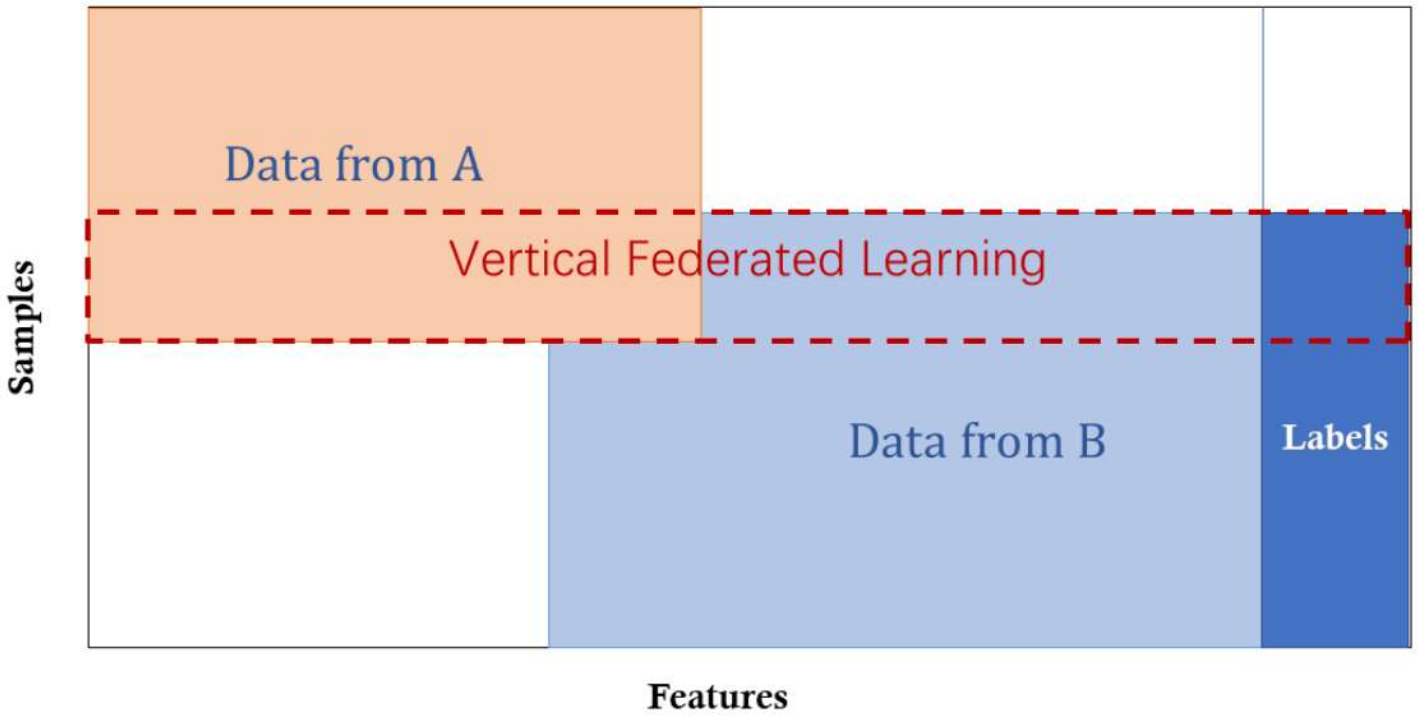
Classification

- Horizontal federated learning



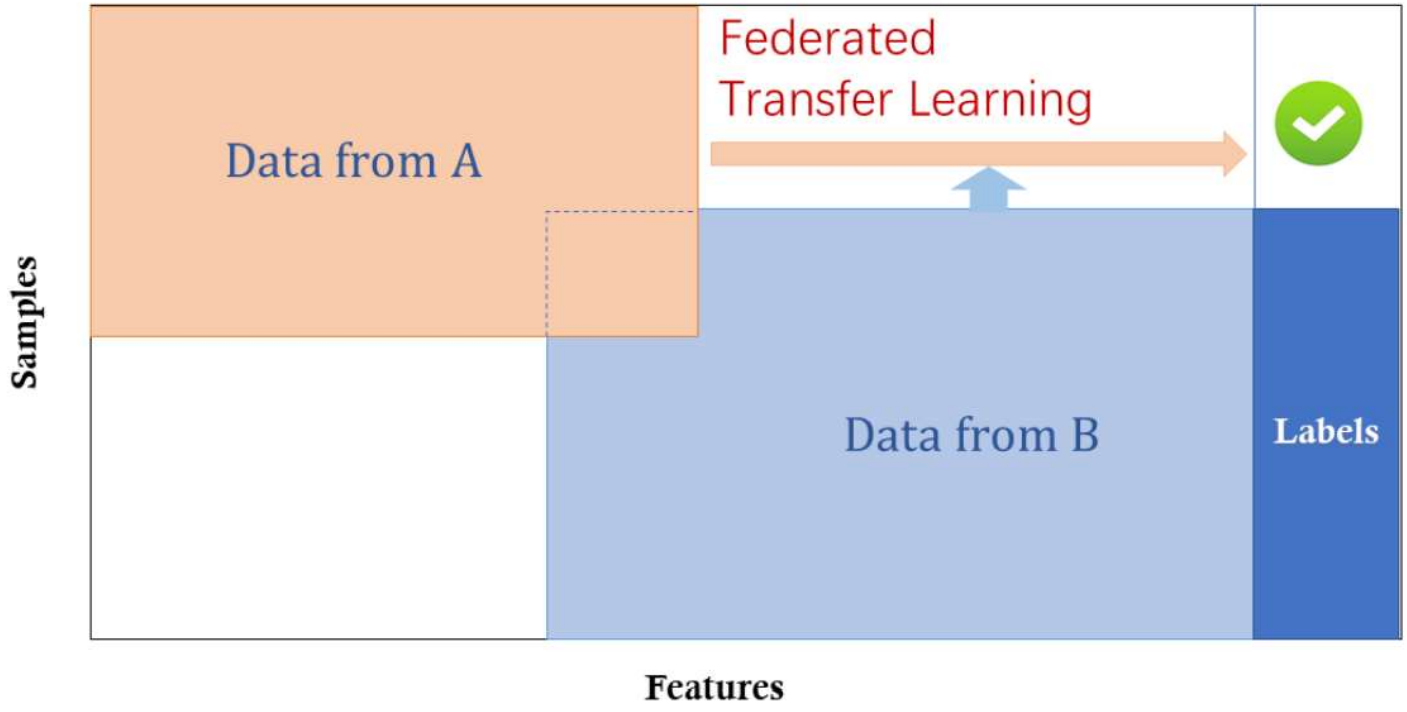
Classification

- Vertical federated learning



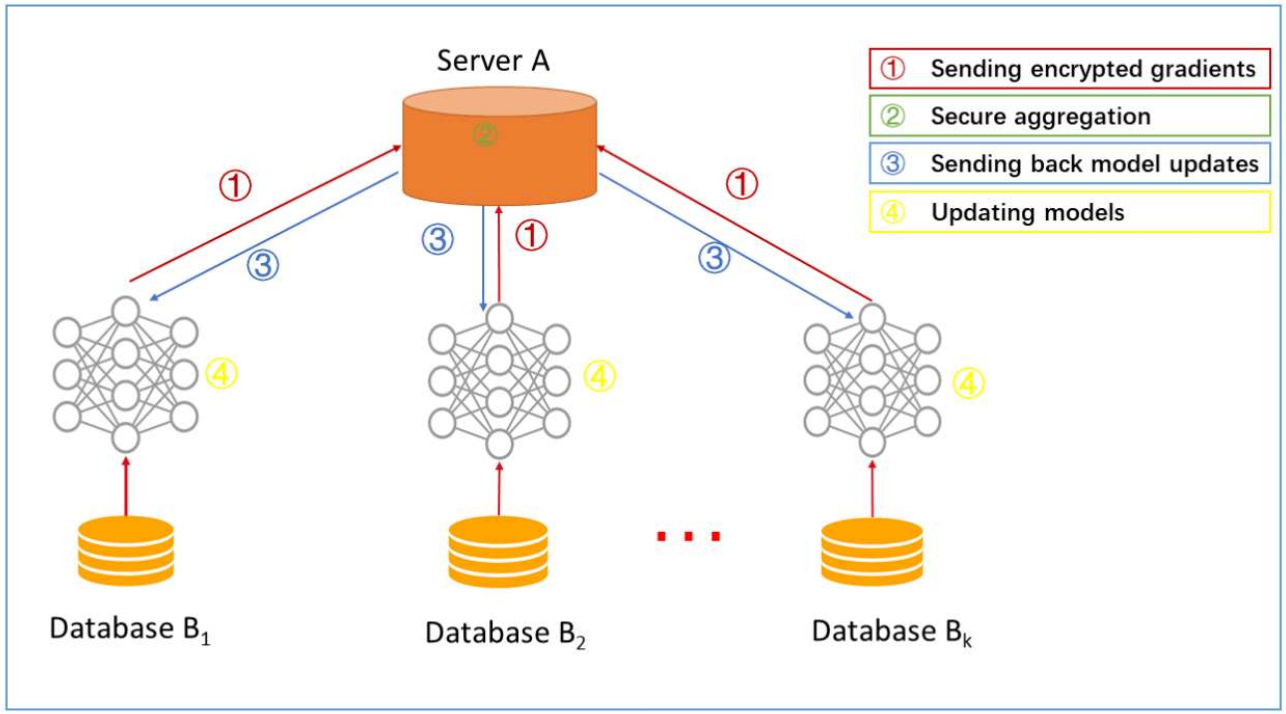
Classification

- Federated Transfer Learning



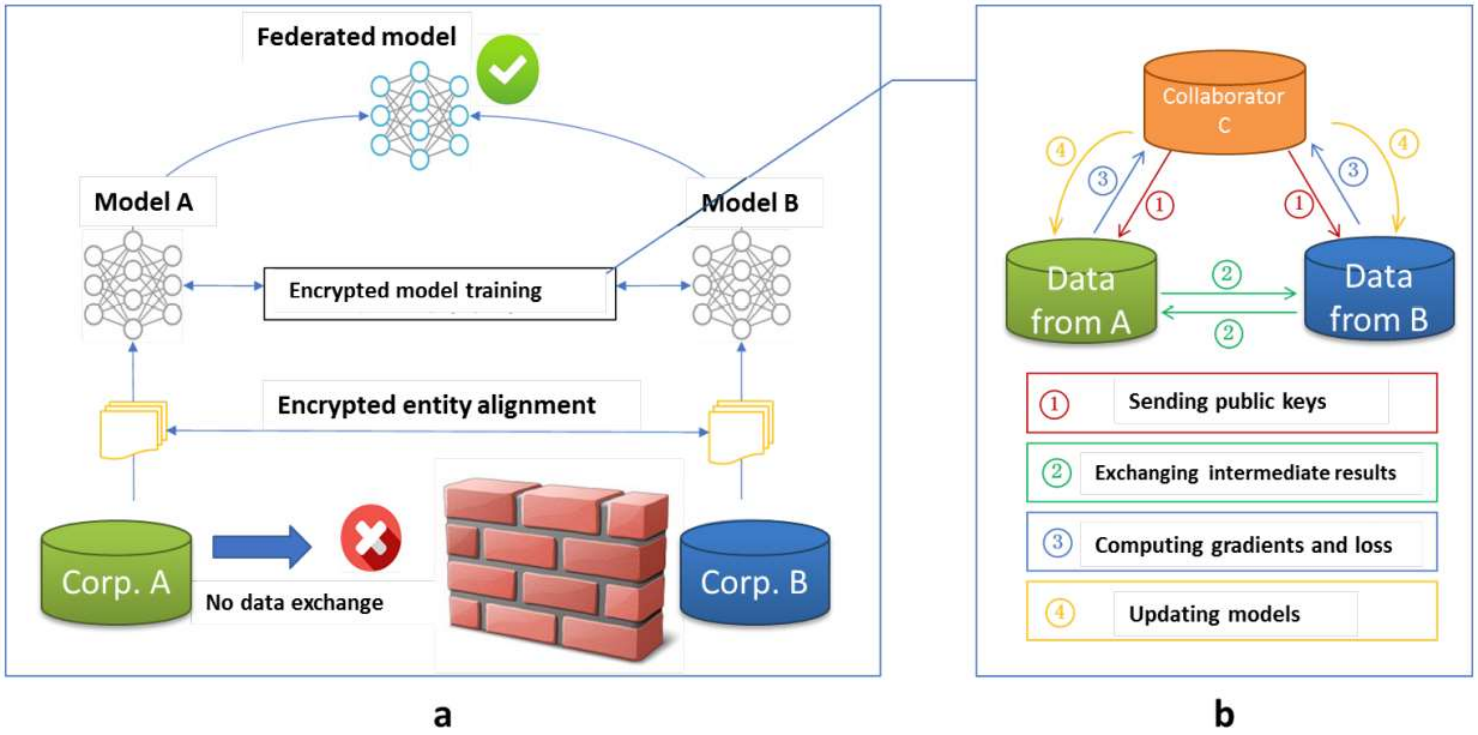
Architecture (synchronous)

- Horizontal federated learning



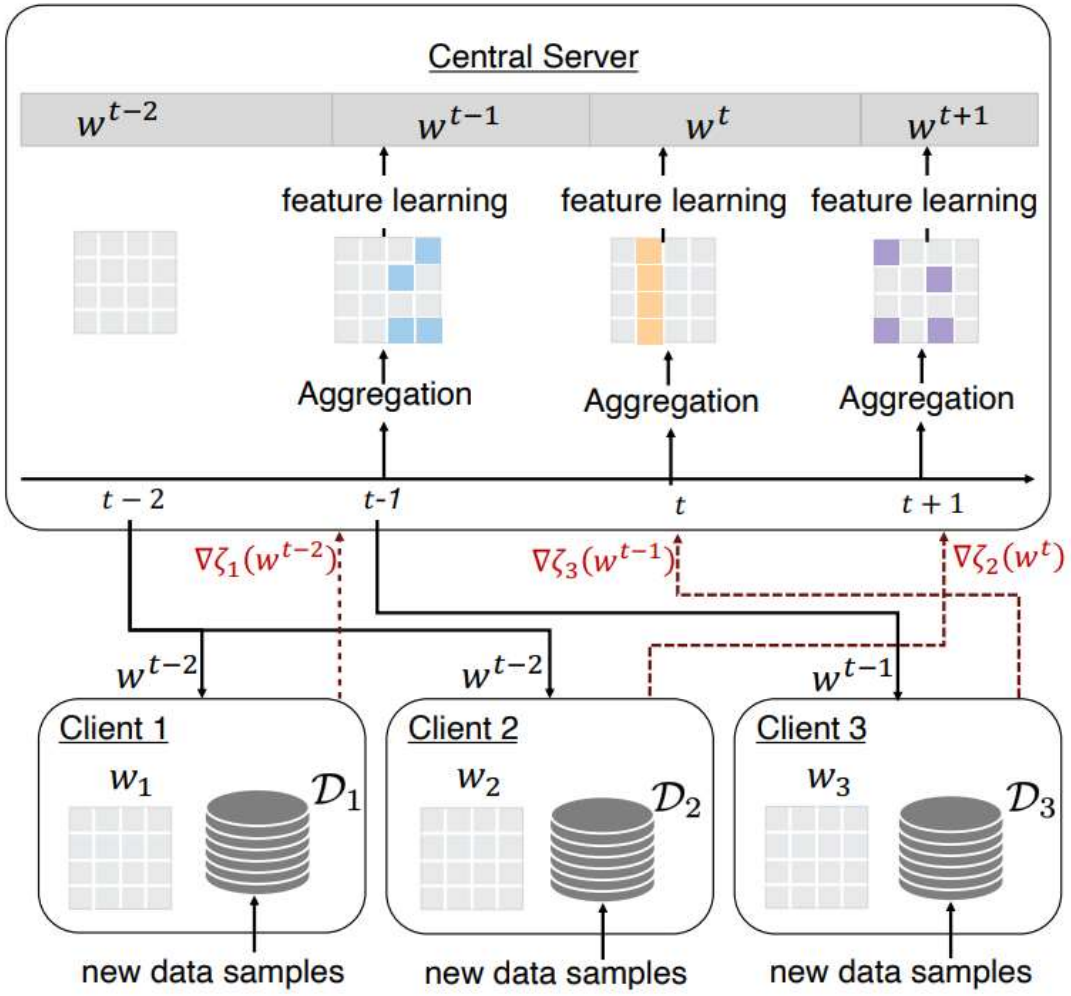
Architecture (synchronous)

- Vertical federated learning

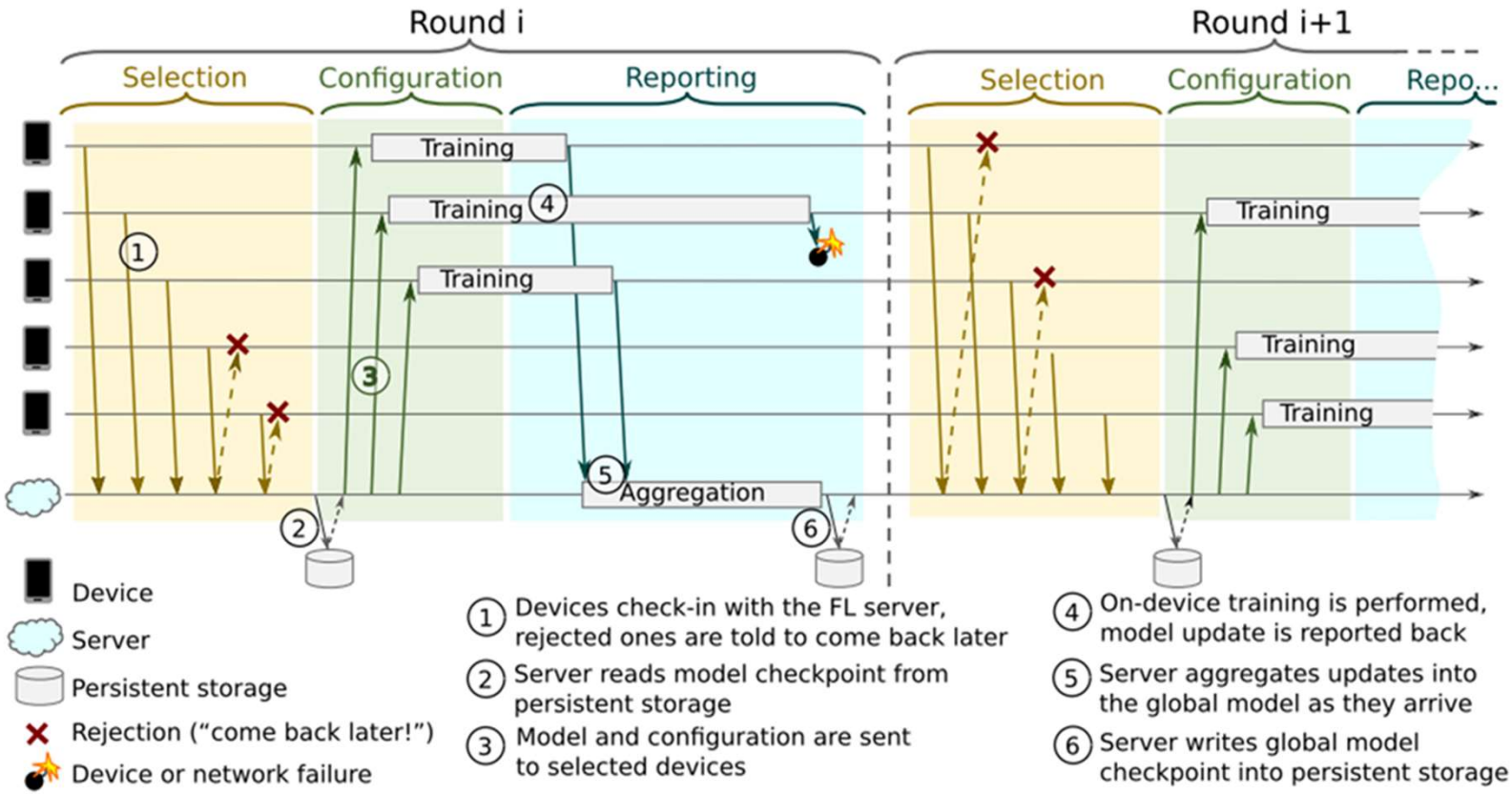


a b

Architecture (asynchronous)



Workflow

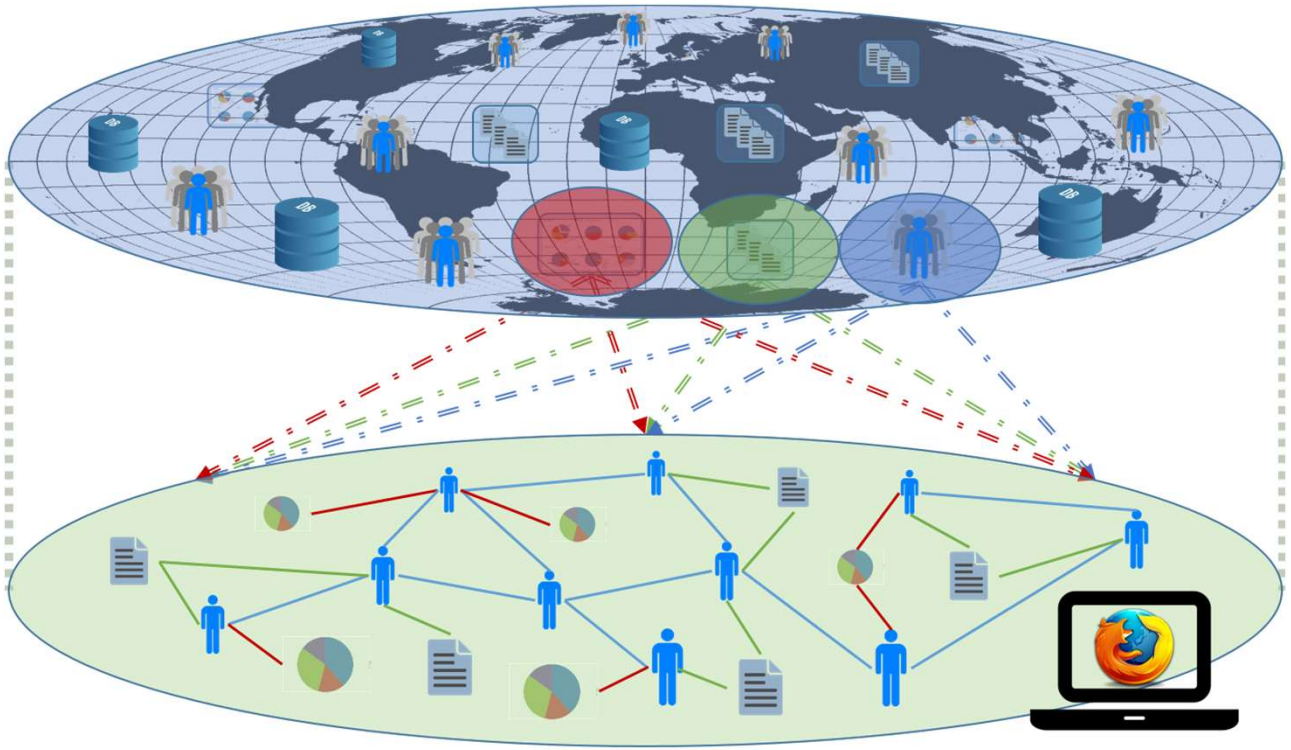

Round i
Round i+1
Selection
Configuration
Reporting
Selection
Configuration
Repo...
Training
Training
Training
Training
Training
Training
Aggregation
1
2
3
4
5
6
Device
Server
Persistent storage
Rejection ("come back later!")
Device or network failure
1 Devices check-in with the FL server, rejected ones are told to come back later
2 Server reads model checkpoint from persistent storage
3 Model and configuration are sent to selected devices
4 On-device training is performed, model update is reported back
5 Server aggregates updates into the global model as they arrive
6 Server writes global model checkpoint into persistent storage

Challenges and opportunities

- Expensive communication.
- Systems heterogeneity.
- Statistical heterogeneity.
- Privacy concerns.

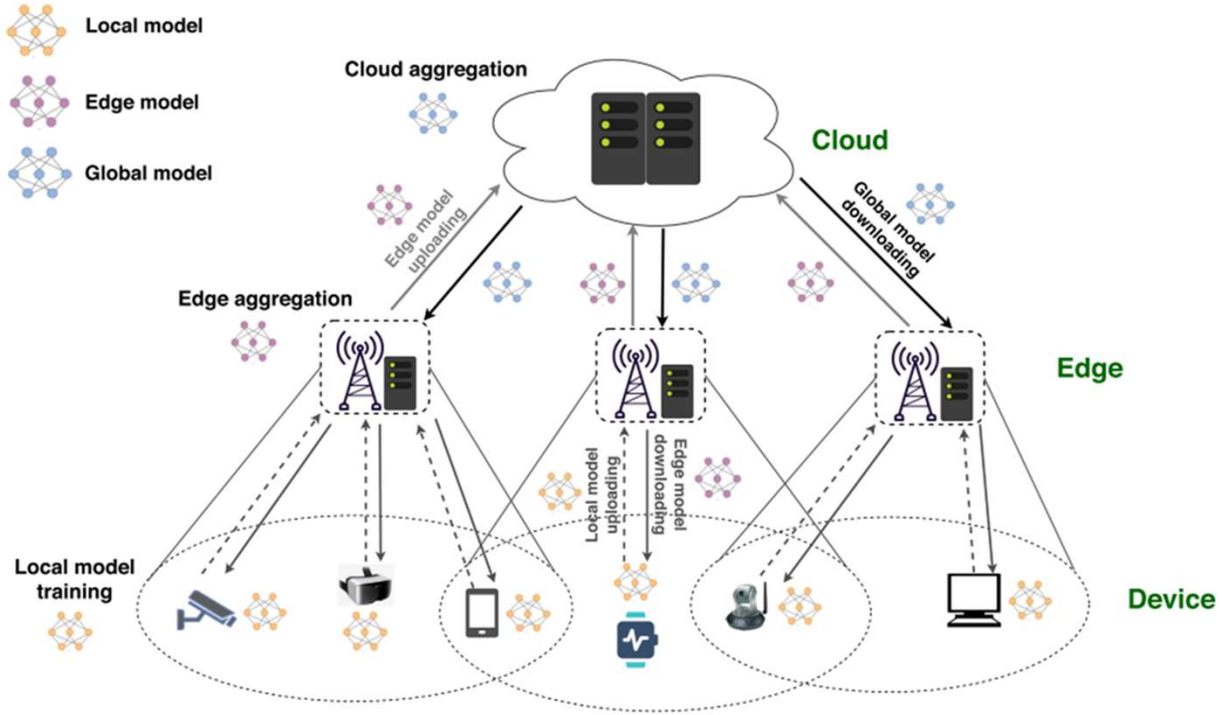
Challenges and opportunities

- Expensive Communication.
  - federated networks are potentially comprised of a massive number of devices, e.g., millions of smart phones, and communication in the network can be slower than local computation by many orders of magnitude.



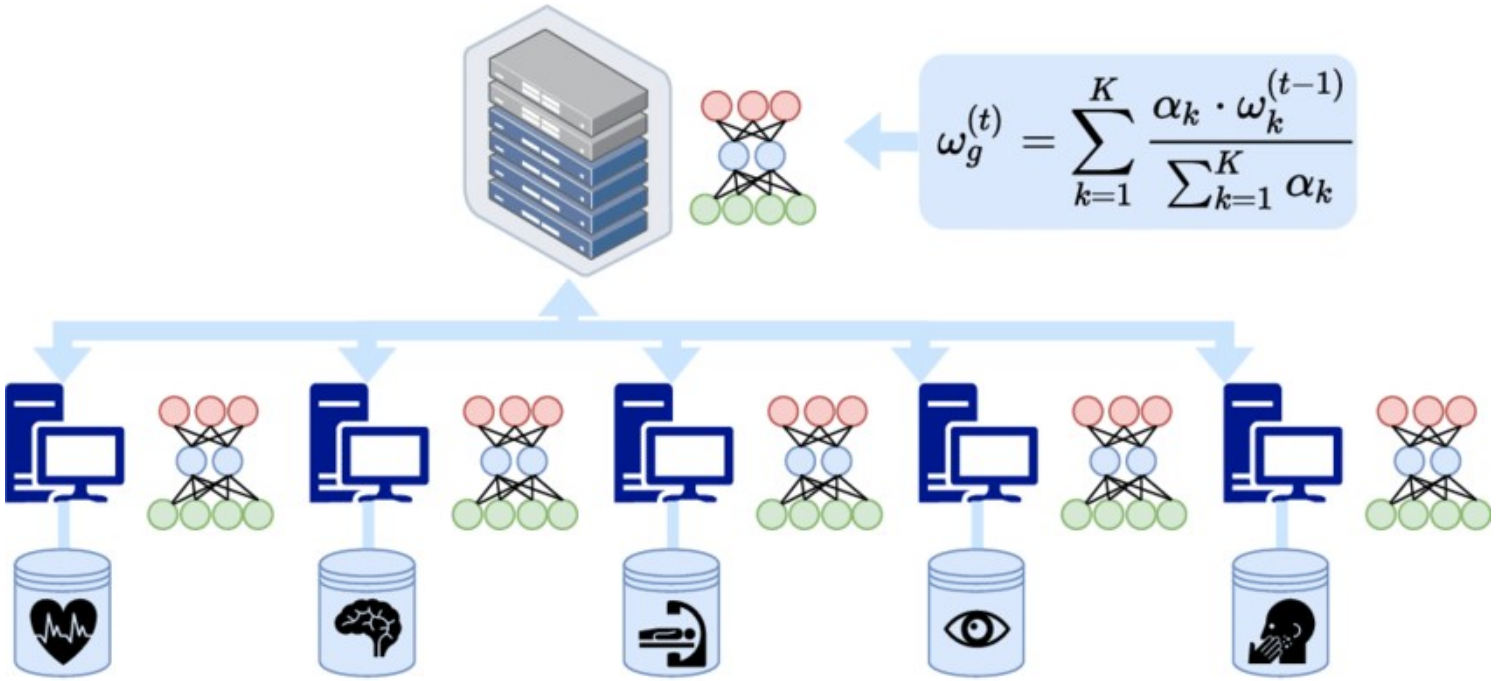
Challenges and opportunities

- Systems Heterogeneity.
  - The storage, computational, and communication capabilities of each device in federated networks may differ due to variability in hardware (CPU, memory), network connectivity (3G, 4G, 5G, wifi), and power (battery level).
  - Each device may also be unreliable.



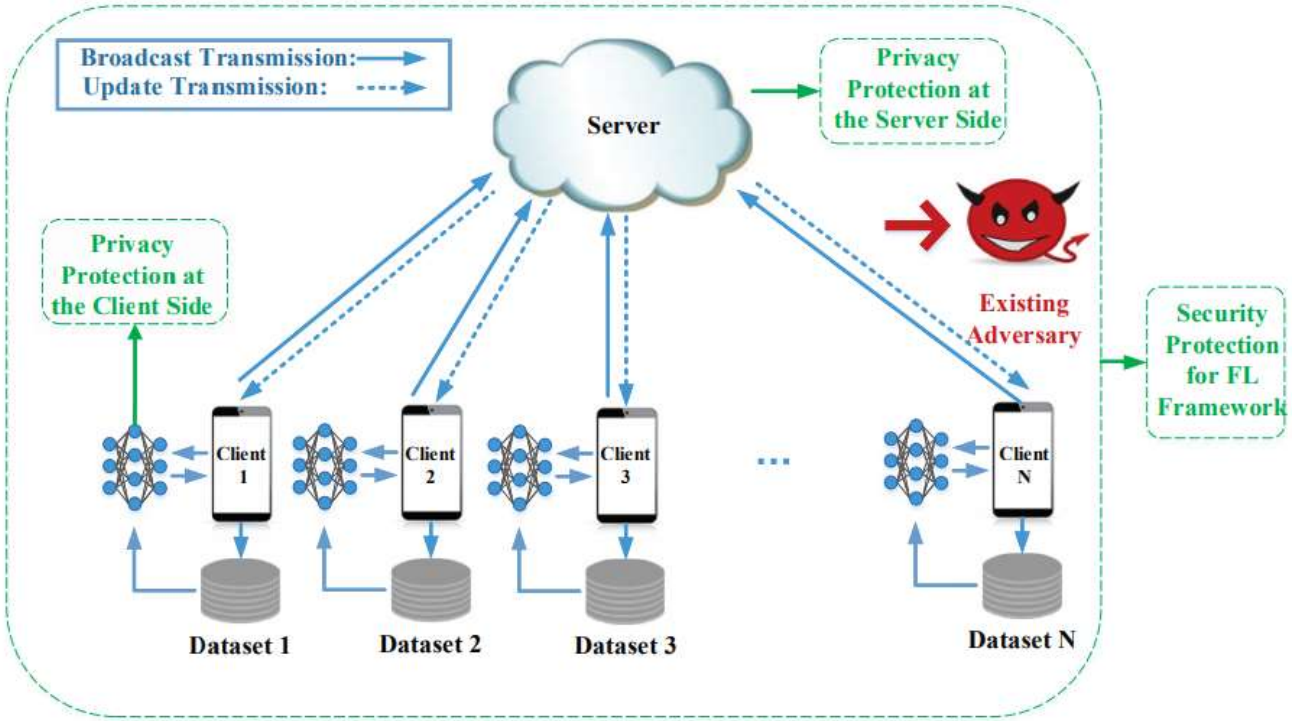
Challenges and opportunities

- Statistical Heterogeneity.
  - Devices frequently generate and collect data in a non-identically distributed manner across the network, e.g., mobile phone users have varied use of language in the context of a next word prediction task.
  - Increases the likelihood of stragglers.



Challenges and opportunities

- Privacy Concerns.
  - communicating model updates throughout the training process can nonetheless reveal sensitive information



Communication-efficiency

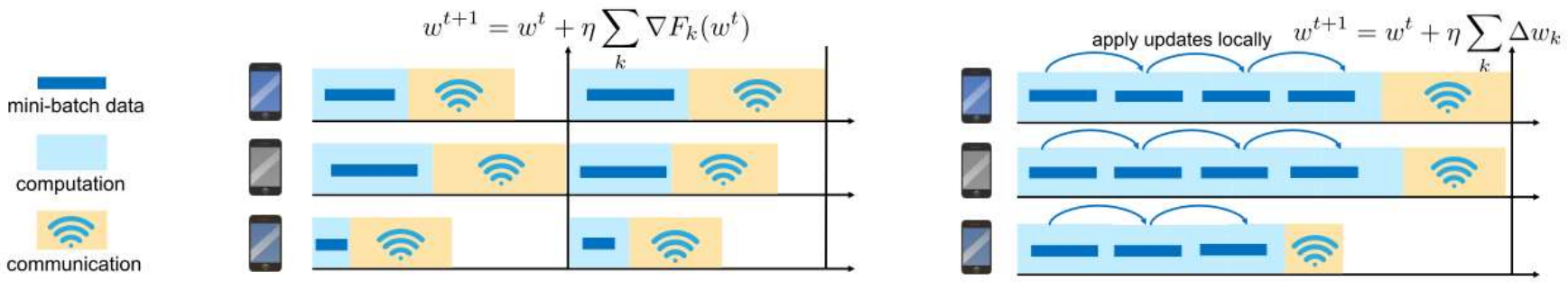
- Local updating
- Compression schemes
- Decentralized training

Local updating

- Mini-batch optimization methods have been shown to have limited flexibility to adapt to communication-computation trade-offs that would maximally leverage distributed data processing.
- Allow for a variable number of local updates to be applied on each machine in parallel at each communication round
- For convex objectives, distributed local-updating primal-dual methods have emerged as a popular way to tackle such a problem.

Local updating

- Left: Distributed (mini-batch) SGD; right: local updating schemes

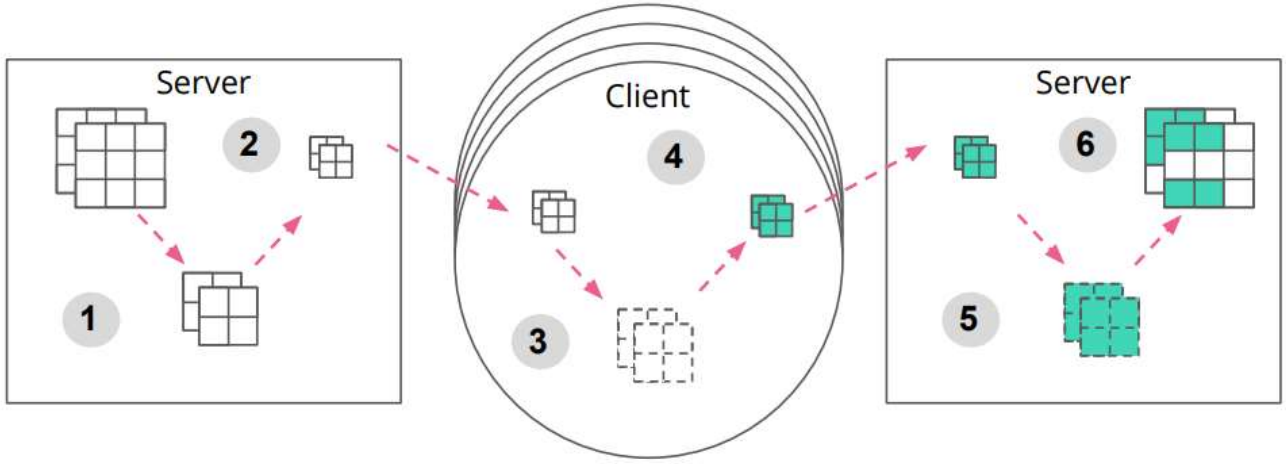


Compression schemes

- Sparsification, subsampling, and quantization can significantly reduce the size of messages communicated at each round.
- In federated environments, conventional approaches face challenges such as low participation of devices.

Compression schemes

- Use lossy compression and dropout to reduce server-to-device communication.

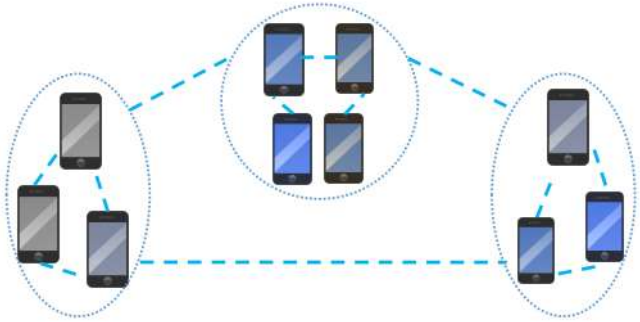


- (1) constructing a sub-model via Federated Dropout, and by (2) lossily compressing the resulting object. This compressed model is then sent to the client, who (3) decompresses and trains it using local data, and (4) compresses the final update. This update is sent back to the server, where it is (5) decompressed and finally, (6) aggregated into the global model

Decentralized Training

- In federated learning, a star network (where a central server is connected to a network of devices) is the predominant communication topology.
- Decentralized algorithms can in theory reduce the high communication cost on the central server.

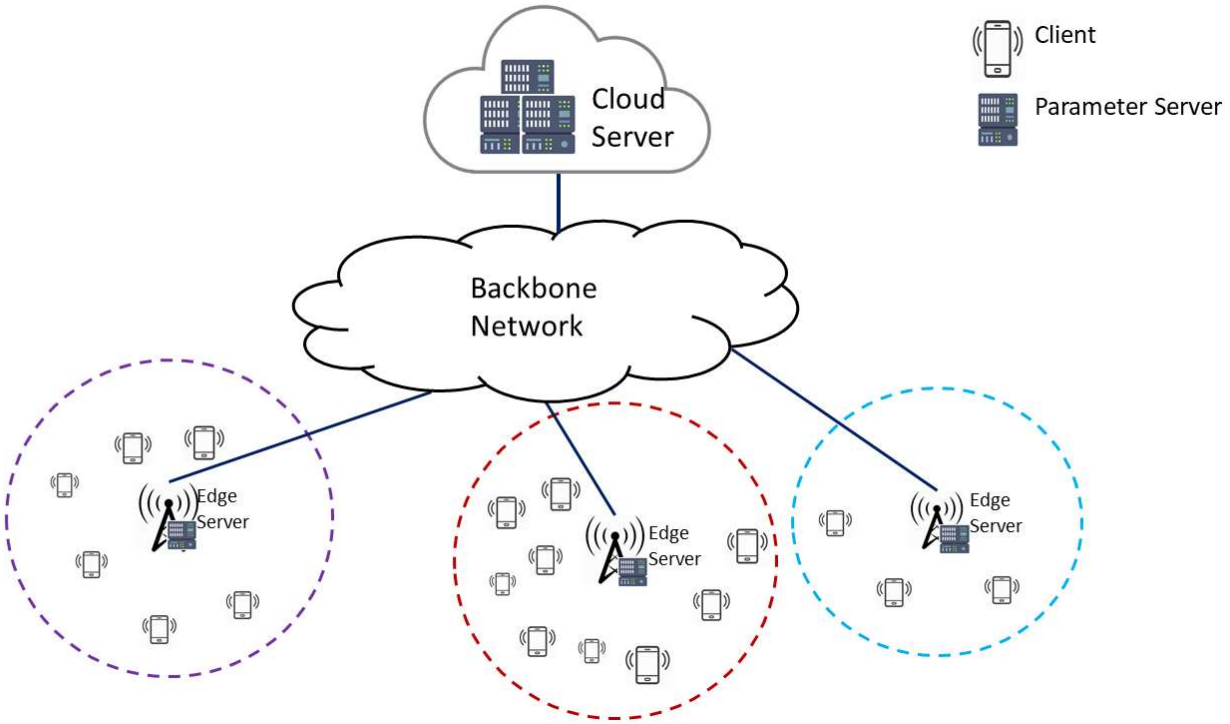
Centralized topology



Decentralized topology

Decentralized training

- Hierarchical communication patterns.



Privacy protection

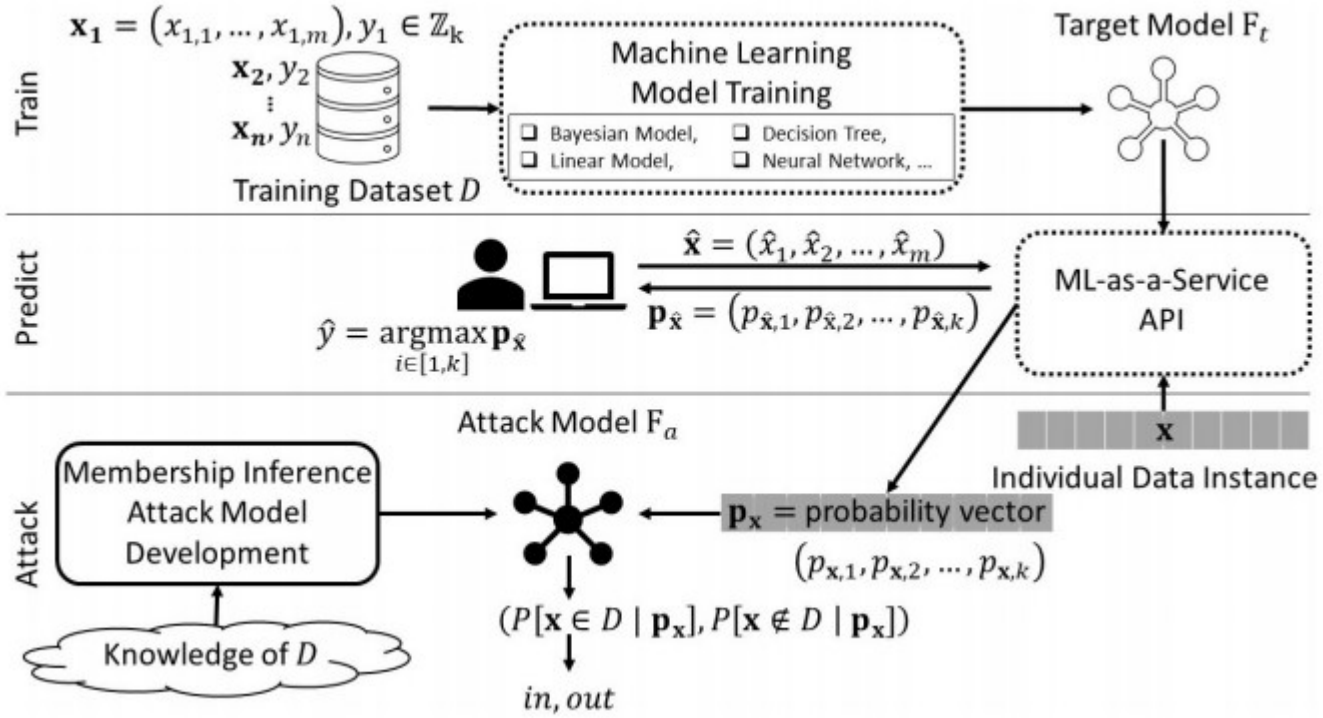
- Privacy threats/attacks in federated learning (FL)
- Enhance the general privacy-preserving feature of FL
- Associated cost with the privacy-preserving techniques

Privacy threats/attacks in FL

- Membership inference attacks
- Unintentional data leakage and reconstruction through inference
- GANs-based inference attacks

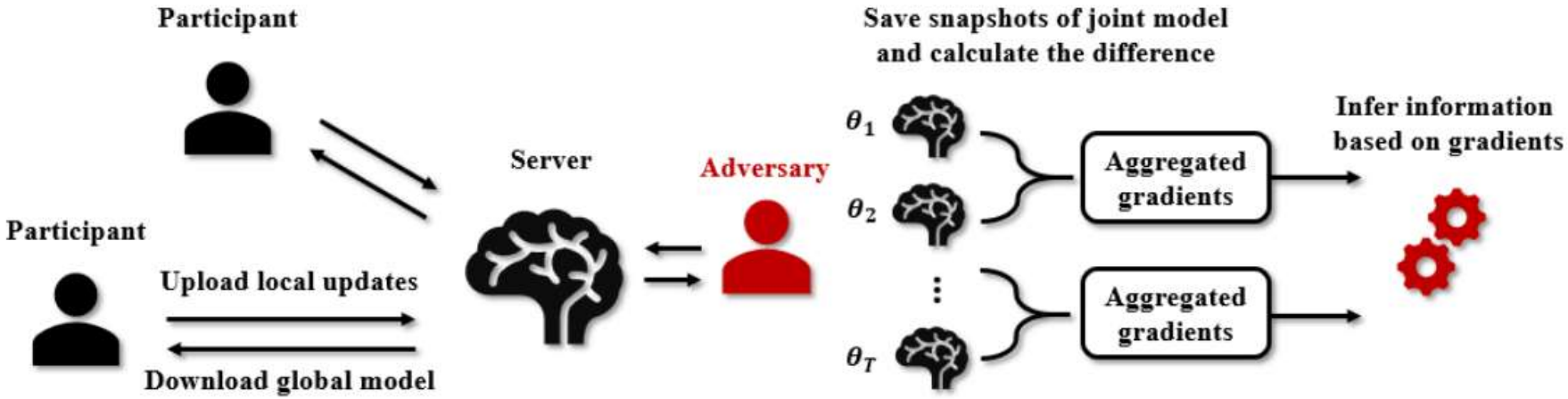
Privacy threats/attacks in FL

- Membership inference attacks
  - The neural network is vulnerable to memorize their training data which is prone to passive and active inference attacks.
  - The attacker misuses the global model to get information on the training data of the other users.



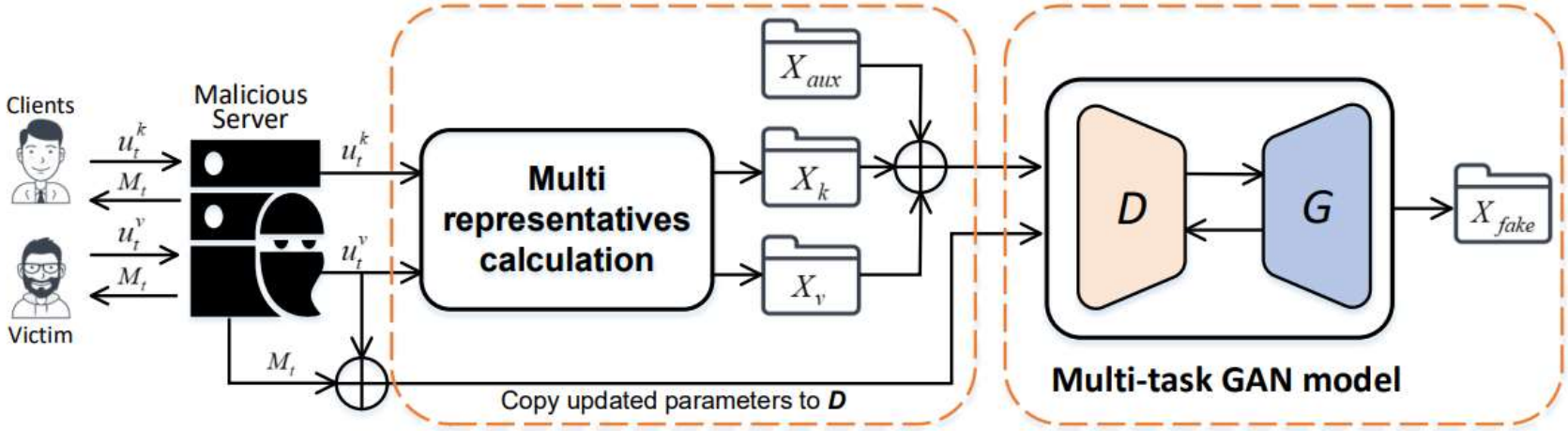
Privacy threats/attacks in FL

- Unintentional data leakage and reconstruction
  - Is a scenario where updates or gradients from clients leak unintended information at the central server.



Privacy threats/attacks in FL

- GANs-based inference attacks
  - GANs are generative adversarial networks that have gained much popularity in big data domains.
  - It is possible to have potential adversaries among FL clients.

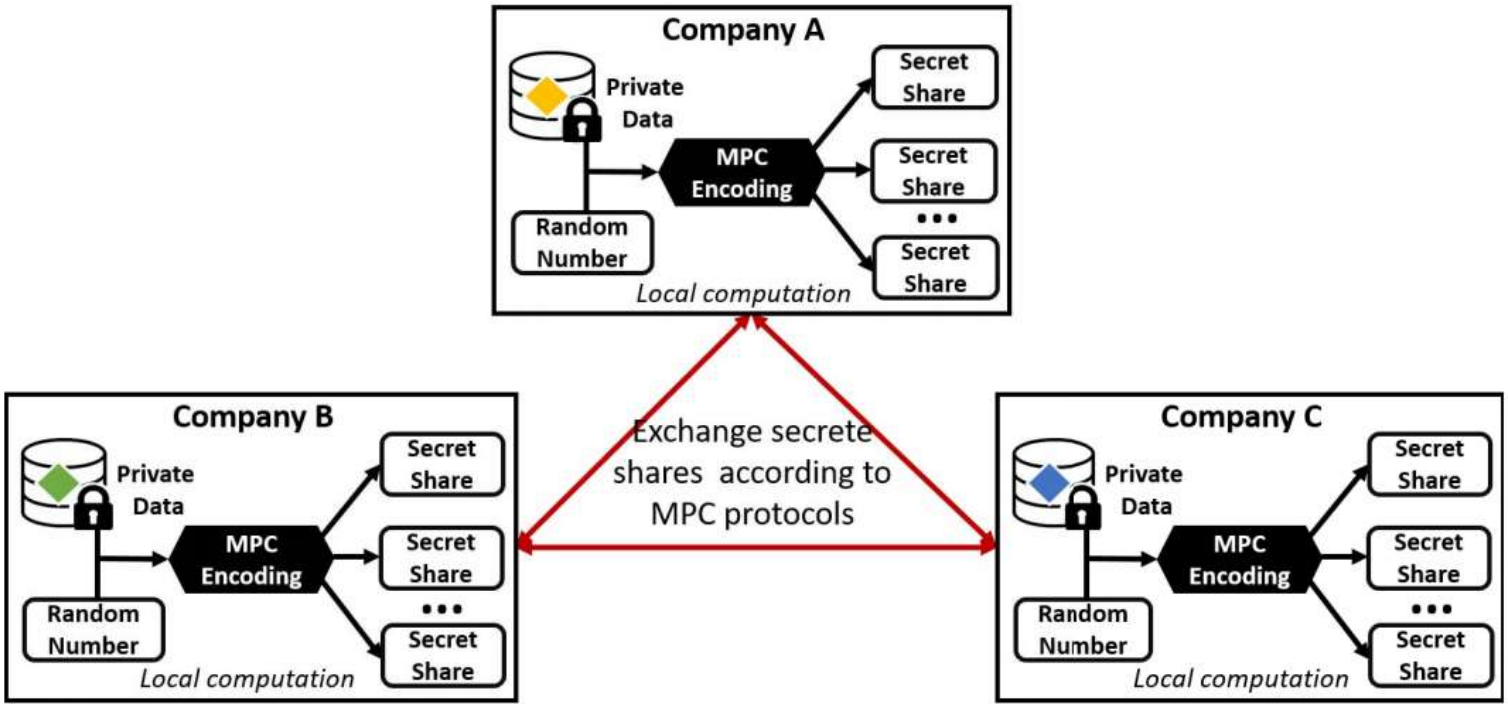


Enhance privacy-preserving in FL

- Secure multi-party computation
- Differential privacy
- VerifyNet
- Adversarial training

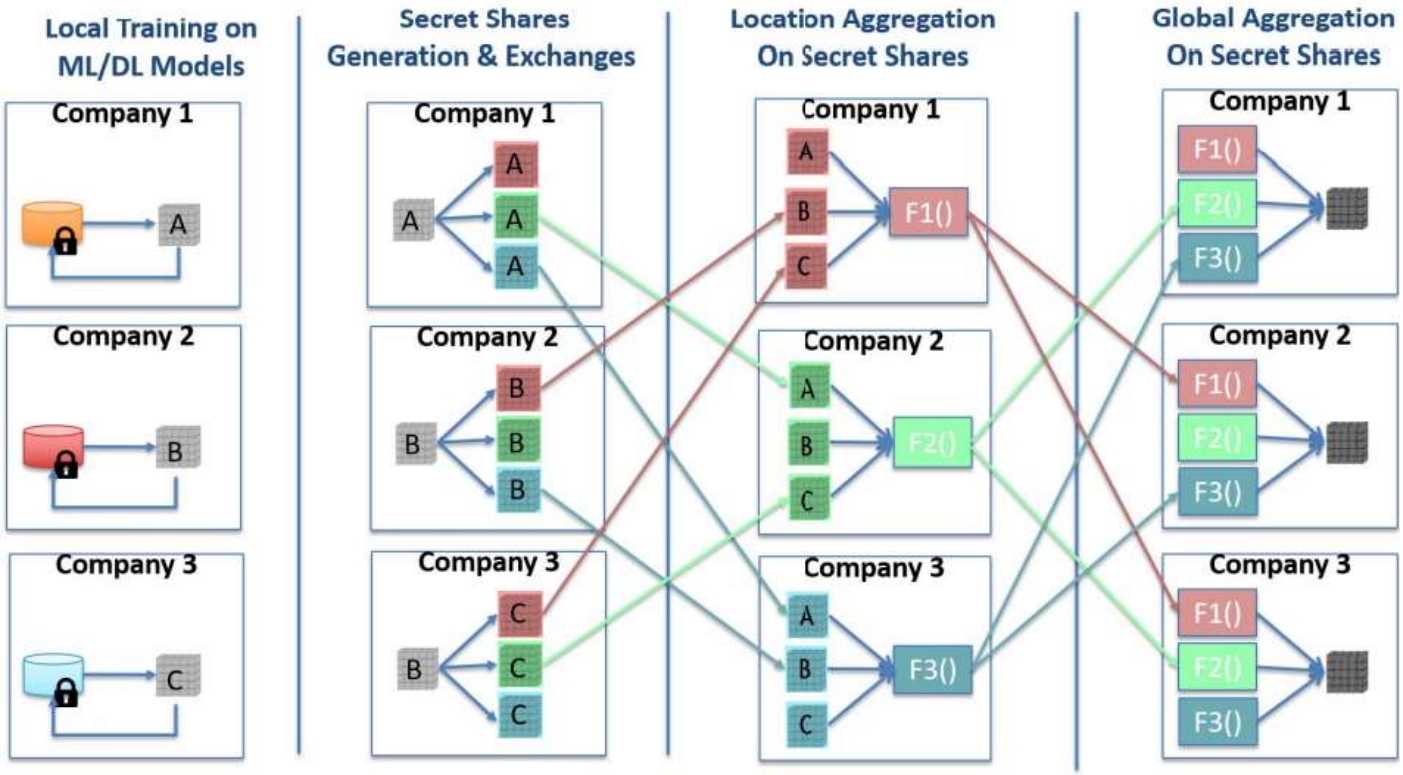
Enhance privacy-preserving in FL

- Secure multi-party computation
  - Secure the inputs of multi-participant while they jointly compute a model or a function.



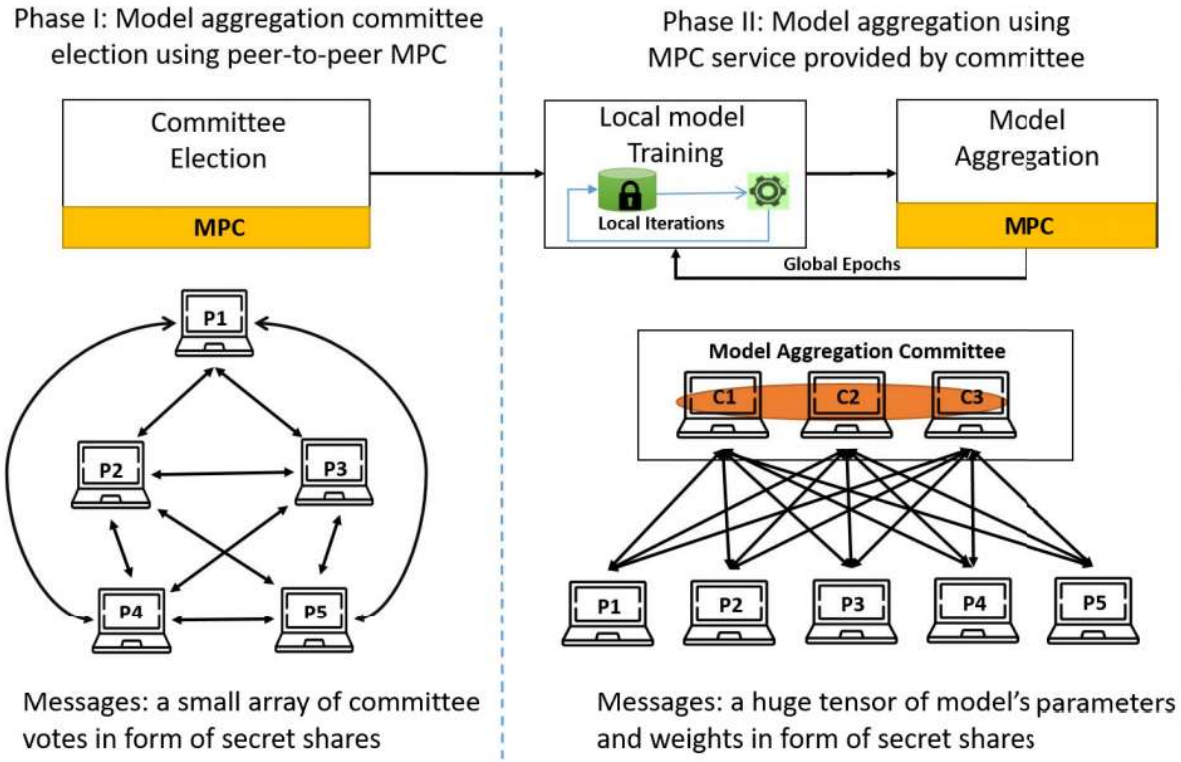
Enhance privacy-preserving in FL

- Secure multi-party computation
  - Secure the inputs of multi-participant while they jointly compute a model or a function.



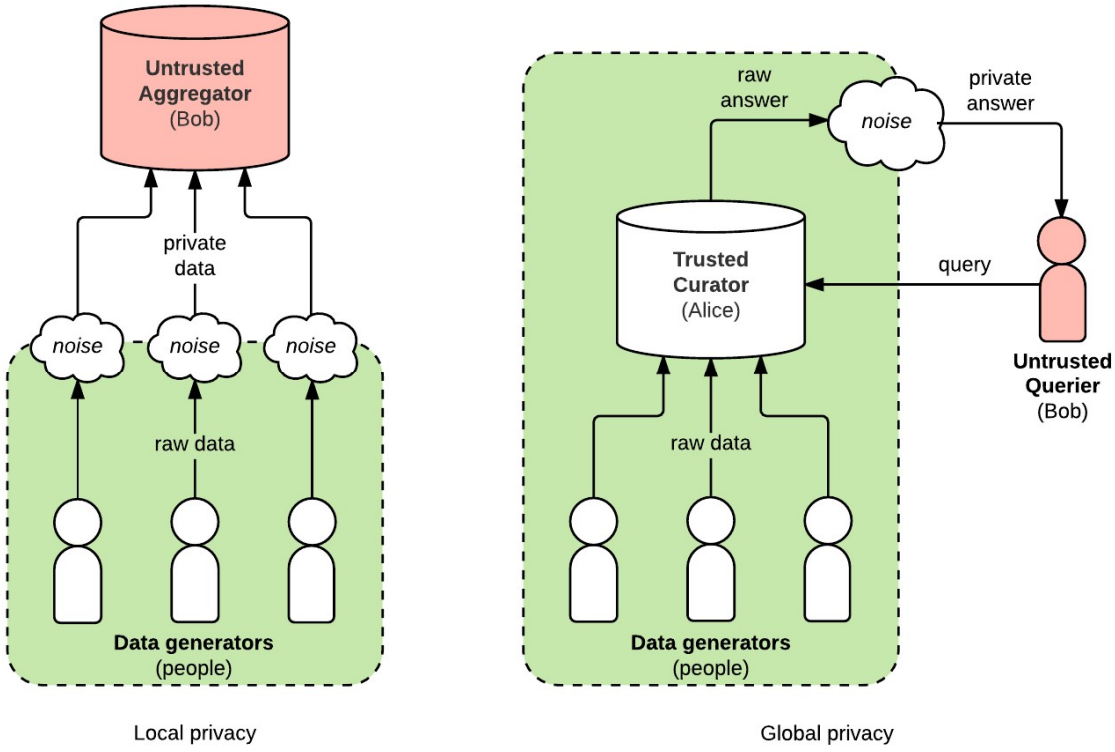
Enhance privacy-preserving in FL

- Secure multi-party computation
  - In FL, the computing efficiency is increased immensely since it only needs to encrypt the parameters instead of the large volume of data inputs.



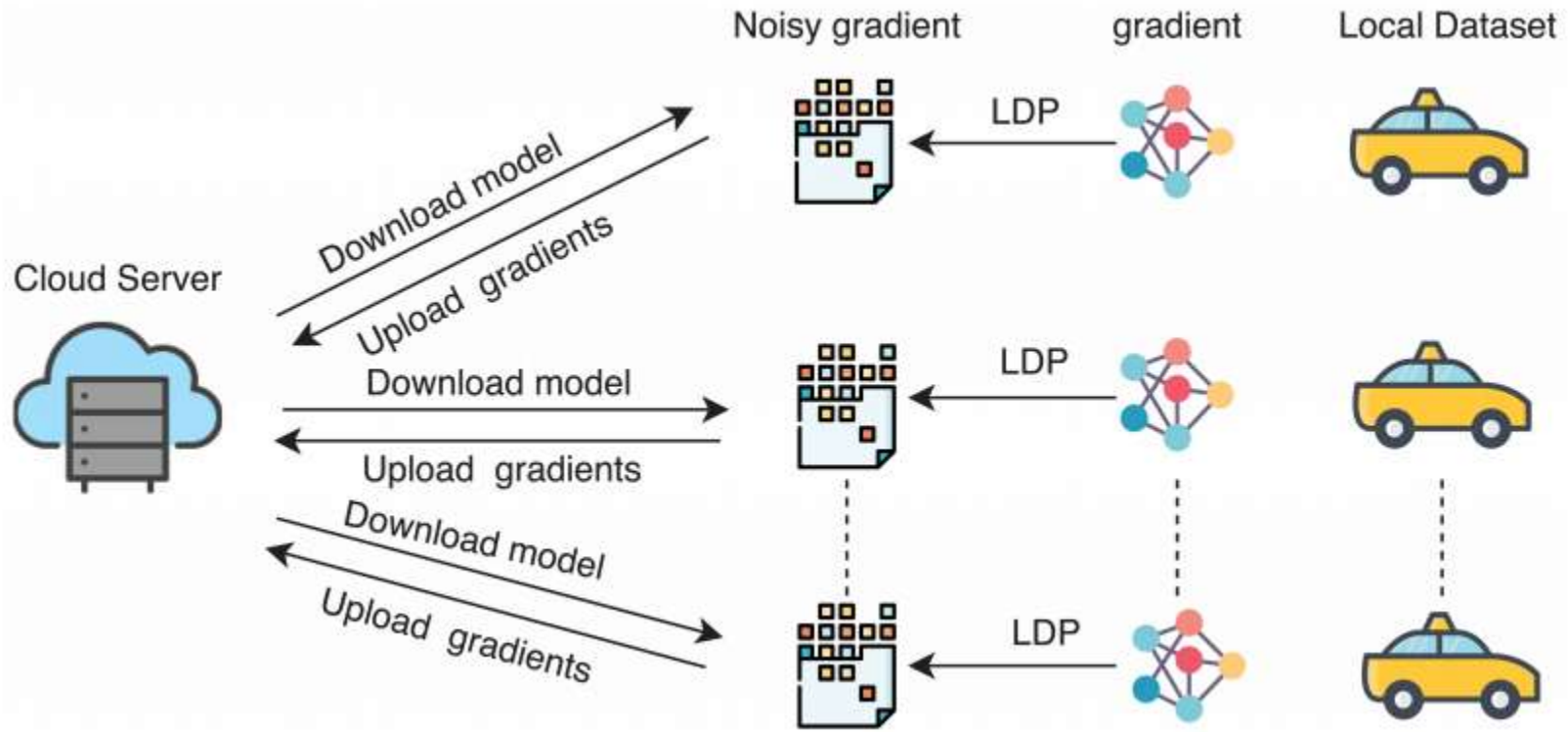
Enhance privacy-preserving in FL

- Differential Privacy
  - Add noise to personal sensitive attributes



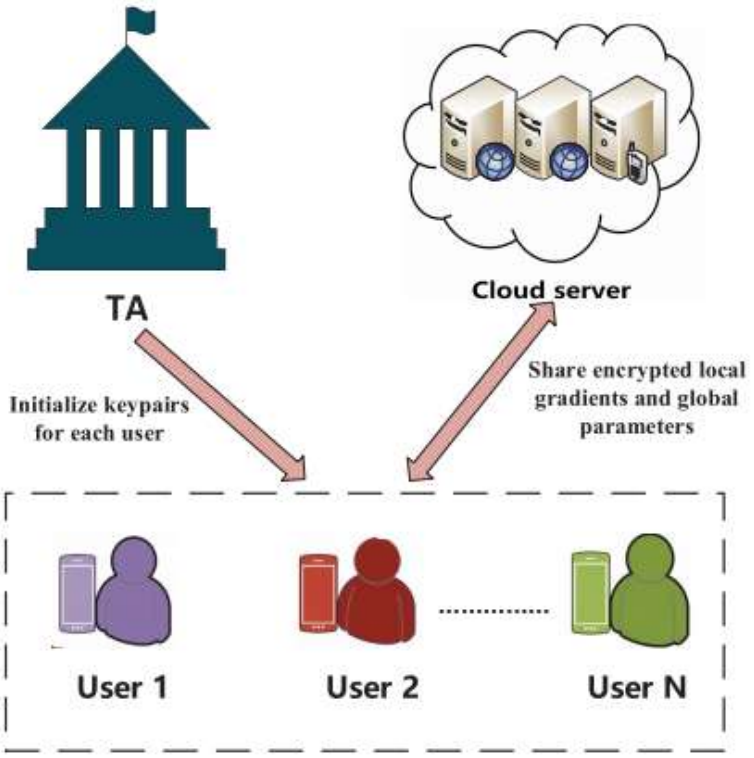
Enhance privacy-preserving in FL

- Differential Privacy
  - DP is introduced to add noise to participants' uploaded parameters



Enhance privacy-preserving in FL

- VerifyNet
  - It gets listed as a preferred mitigation strategy to preserve privacy as it provides double-masking protocol which makes it difficult for attackers to infer training data.

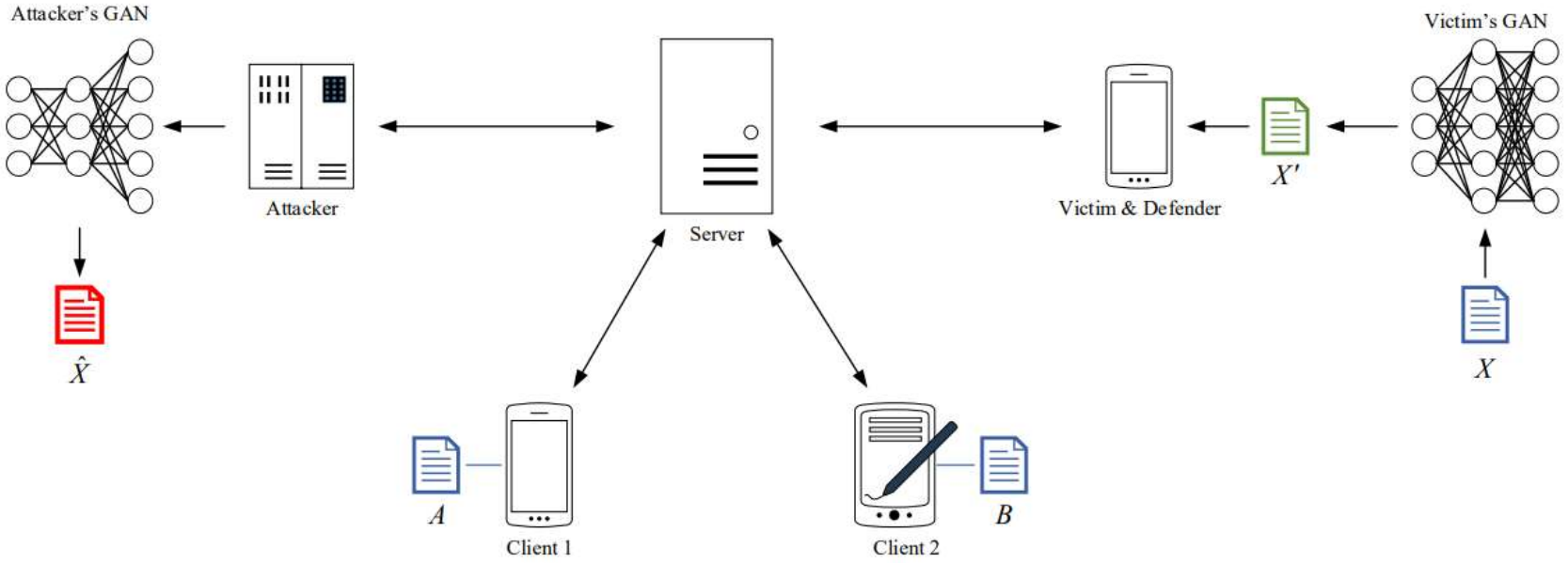


Enhance privacy-preserving in FL

- Adversarial training
  - Evasion attacks from an adversarial user aims to fool ML models by injecting adversarial samples into the machine learning models.
  - The attacker tries to impact the robustness of the FL model with perturbed data.
  - Adversarial training, which is a proactive defense technique, tries all permutations of an attack from the beginning of the training phase to make the FL global model robust to known adversarial attacks.

Enhance privacy-preserving in FL

- Adversarial training
  - Use GAN to generate fake training data.



Associated cost

Approach	Cost	Methodology
Secure Multi-party Computation	Efficiency loss due to encryption	Encrypt uploaded parameters
Differential Privacy	Accuracy loss due to added noise in client's model	Add random noise to uploaded parameters
Hybrid	Subdued cost on both efficiency and accuracy	Encrypt the manipulated parameter
VerifyNet	Communication overhead	Double-masking protocol Verifiable aggregation results
Adversarial Training	Computation power, training time for adversarial samples	Include adversarial samples in training data

Reference

- https://analyticsindiamag.com/top-tools-distributed-machine-learning-tensorflow/
- https://www.sciencedirect.com/science/article/pii/S0167739X18319903?via%3Dihub
- https://ieeexplore.ieee.org/abstract/document/8123913/
- https://www.ibm.com/downloads/cas/ZBZLY7KL?_ga=2.148238199.1762516747.1577395260-1128561362.1577395260
- https://unctad.org/en/Pages/DTL/STI_and_ICTs/ICT4D-Legislation/eCom-Data-Protection-Laws.aspx
- https://arxiv.org/pdf/1902.04885.pdf
- https://arxiv.org/pdf/1911.02134.pdf
- https://arxiv.org/abs/1902.01046
- https://arxiv.org/abs/1908.07873
- https://arxiv.org/abs/2002.11343
- https://human-centered.ai/secure-federated-machine-learning/
- https://www.researchgate.net/publication/343735312_Inverse_Distance_Aggregation_for_Federated_Learning_with_Non-IID_Data
- https://arxiv.org/pdf/1909.06512v2.pdf
- https://arxiv.org/pdf/1812.07210.pdf
- https://www.catalyzex.com/paper/arxiv:1905.06641
- https://www.sciencedirect.com/science/article/pii/S0167739X20329848
- https://arxiv.org/pdf/1805.04049.pdf
- https://arxiv.org/pdf/1812.00535.pdf
- https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9139658
- https://www.accessnow.org/understanding-differential-privacy-matters-digital-rights/
- https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9253545
- https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8765347
- https://arxiv.org/pdf/2004.12571.pdf

谢谢！