Heterogeneity in Federated Learning

Jiaqi Wang & Fenglong Ma

College of Information Sciences and Technology

The Pennsylvania State University

jqwang@psu.edu, fenglong@psu.edu

PennState

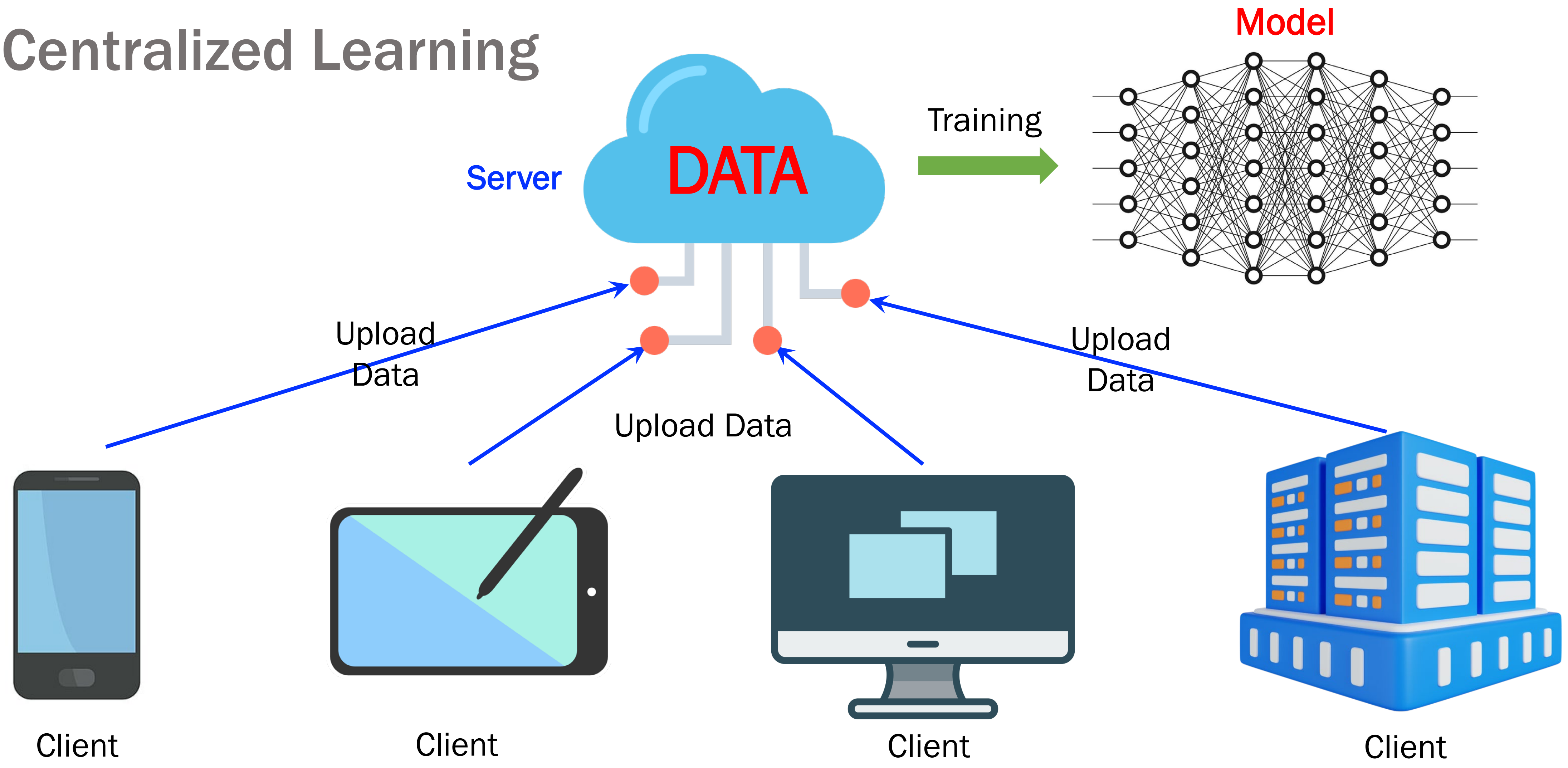
Content

- Part 1: Federated Learning Introduction
- Part 2: Data/Statistical Heterogeneity
- Part 3: Model Heterogeneity
- Part 4: System Heterogeneity
- Part 5: Conclusion and Future Work

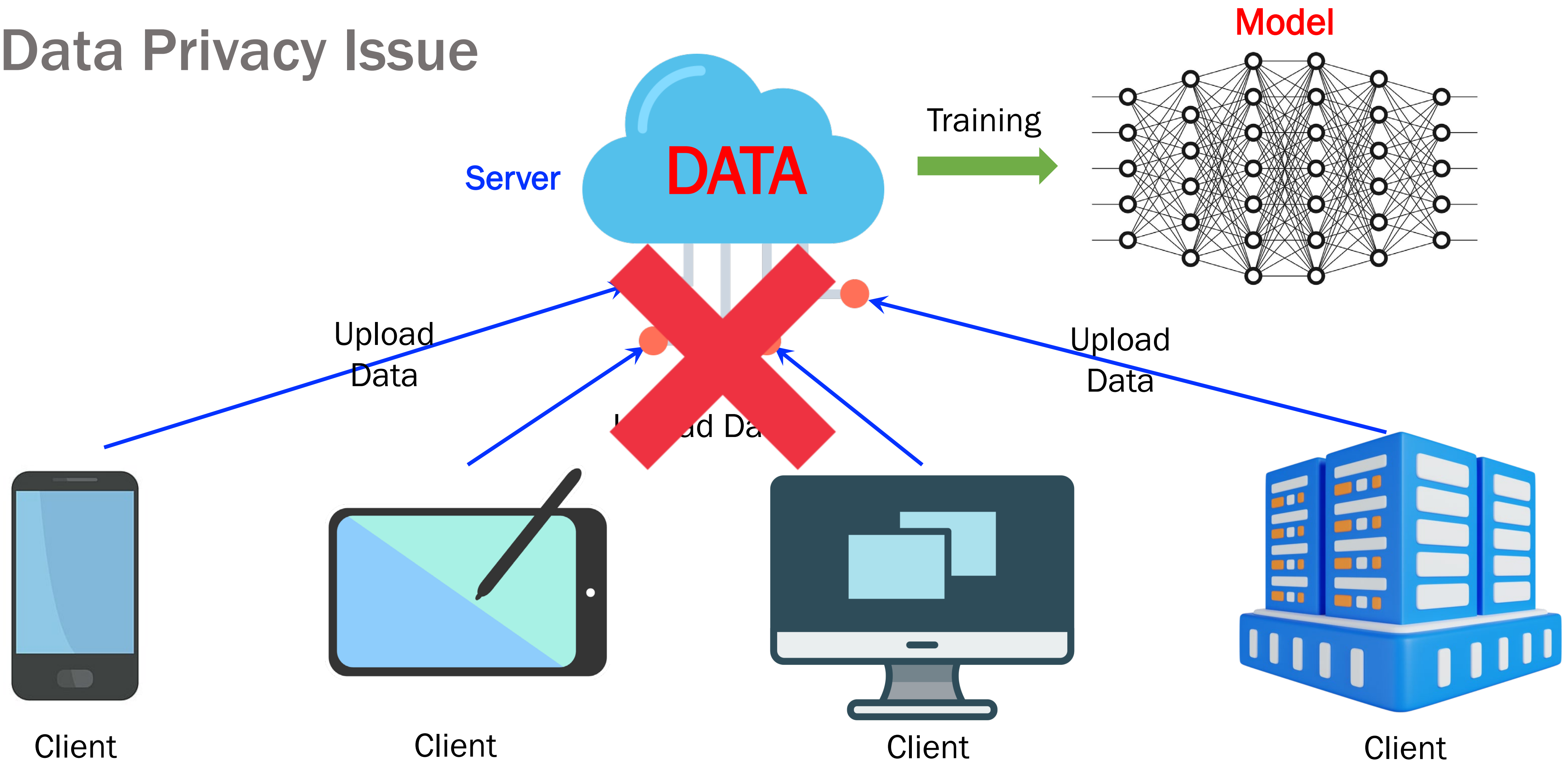
Part 1

- Part 1: Federated Learning Introduction
- Part 2: Data/Statistical Heterogeneity
- Part 3: Model Heterogeneity
- Part 4: System Heterogeneity
- Part 5: Conclusion and Future Work

Centralized Learning

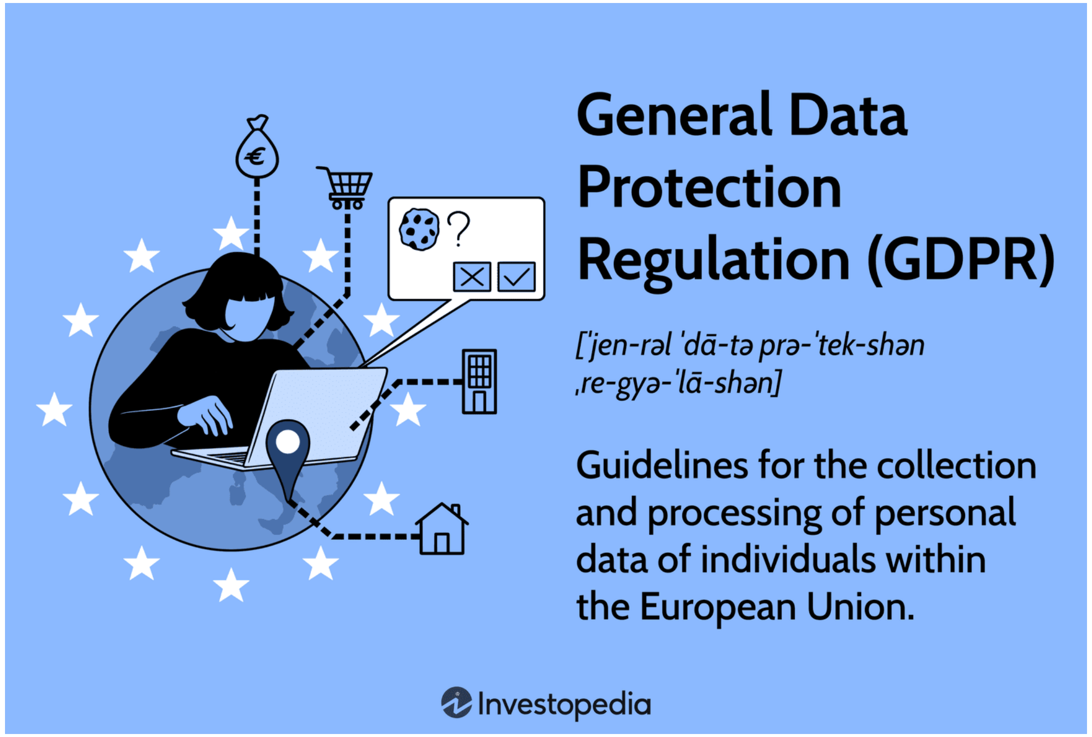


Data Privacy Issue



Data Privacy Laws

Health Insurance Portability and Accountability Act



General Data Protection Regulation (GDPR)

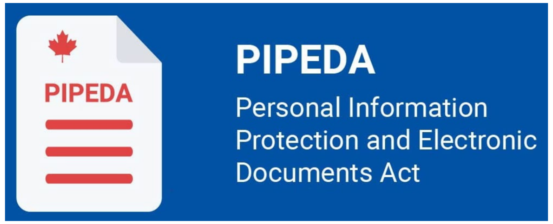
[ˈjen-rəl ˈdā-tə prə-ˈtek-shən ˌre-gyə-ˈlā-shən]

Guidelines for the collection and processing of personal data of individuals within the European Union.

Investopedia

COPPA

Children's Online Privacy Protection Act



PIPEDA

Personal Information Protection and Electronic Documents Act

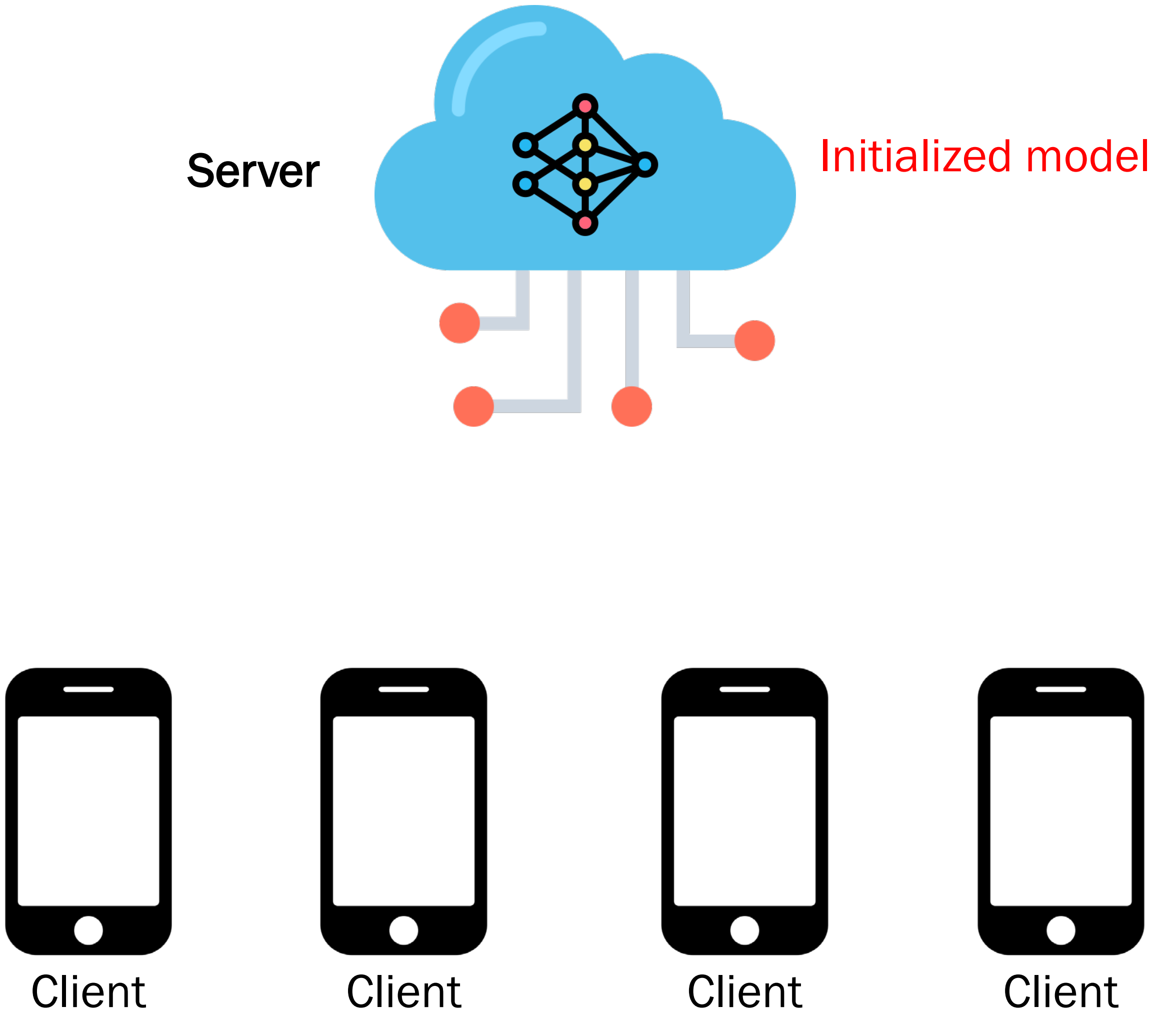
China's Draft Personal Information Protection Law (PIPL)

OVERVIEW OF THE PRIVACY ACT OF 1974

2020 EDITION
UNITED STATES DEPARTMENT OF JUSTICE

Federated Learning

- Federated Learning (FL) aims to collaboratively train a machine learning (ML) model while keep the data decentralized.


Server

Initialized model

Client

Client

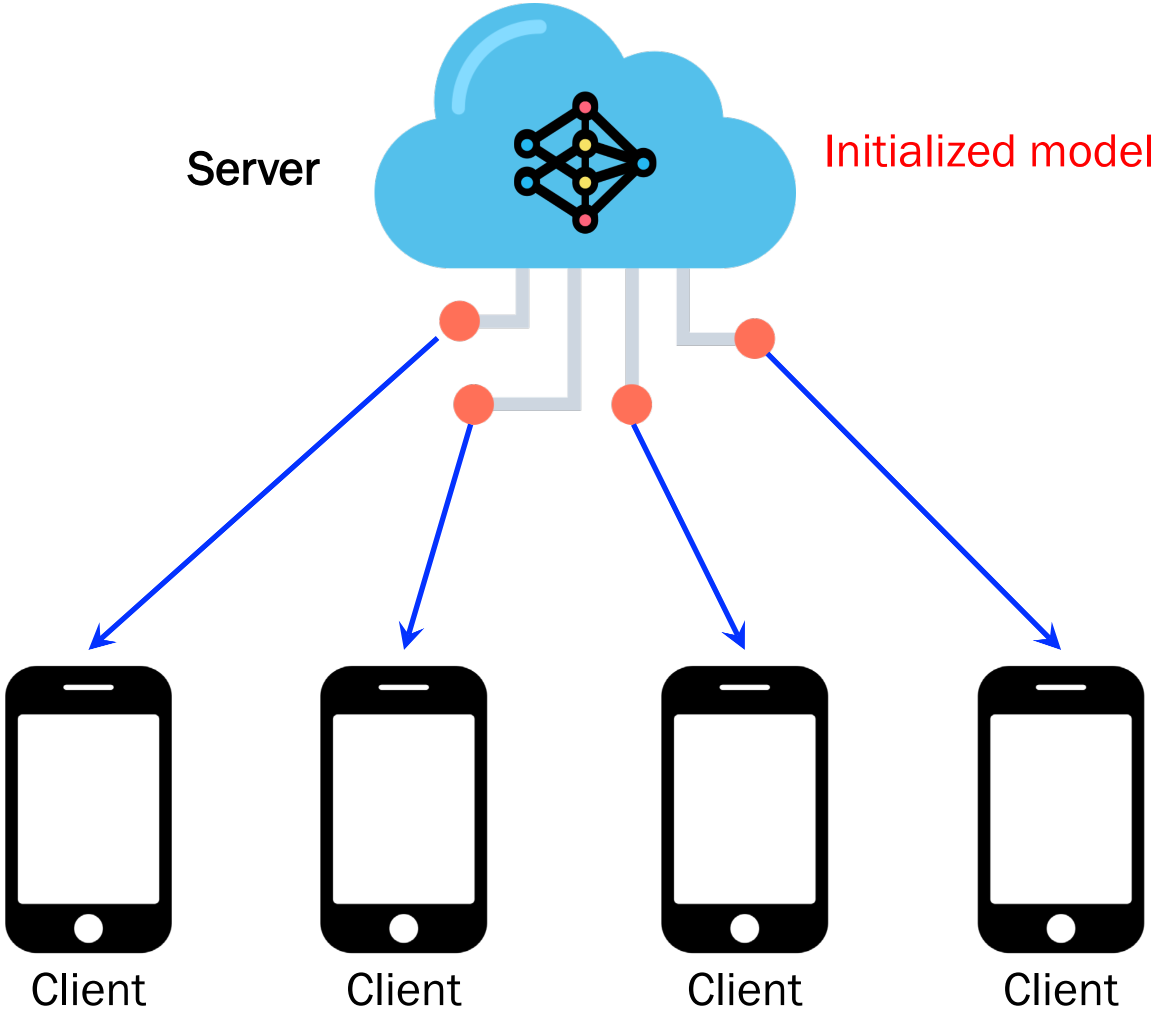
Client

Client

McMahan et al. "Communication-efficient learning of deep networks from decentralized data." Artificial intelligence and statistics. PMLR, 2017.

Federated Learning

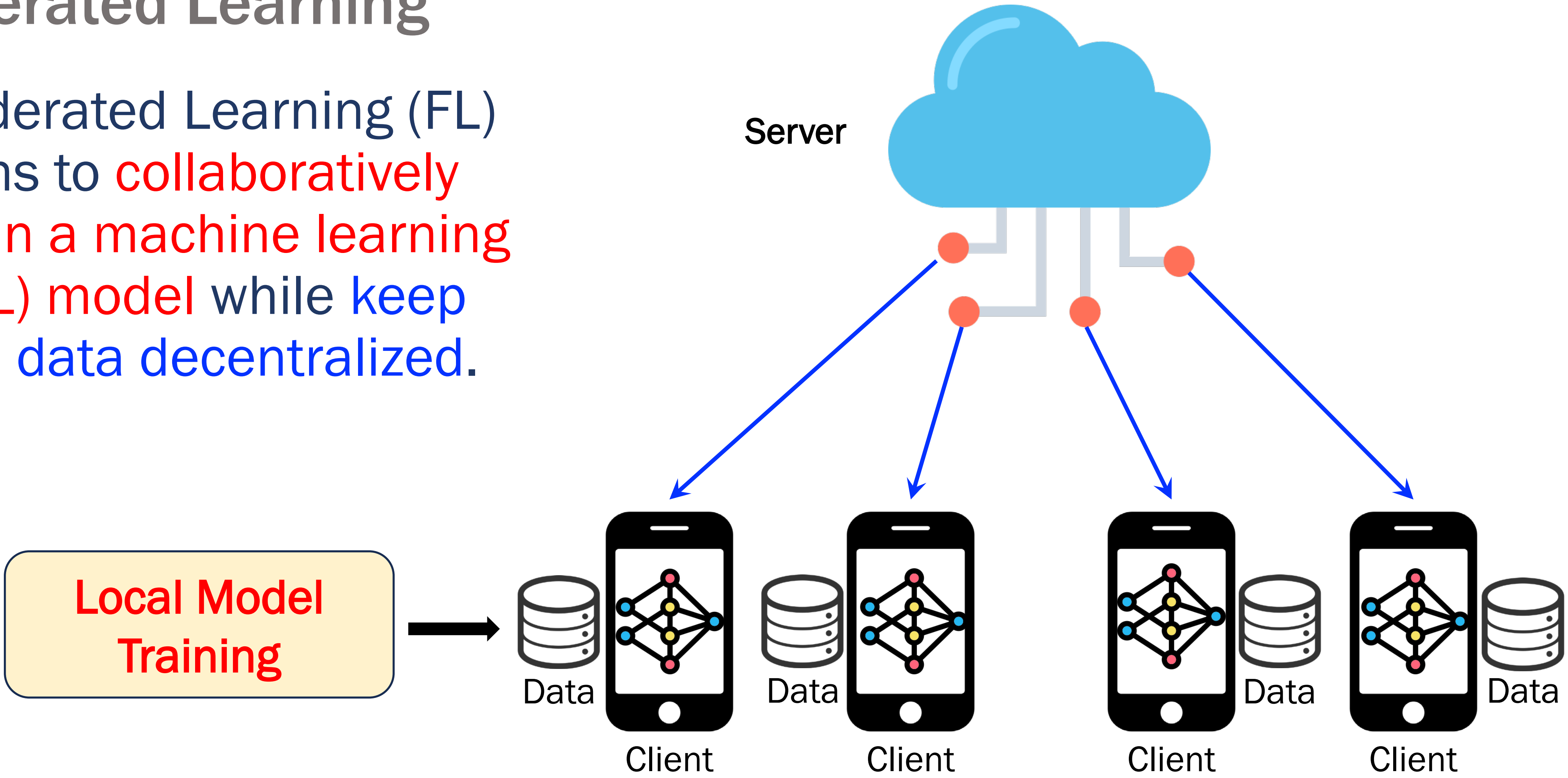
- Federated Learning (FL) aims to collaboratively train a machine learning (ML) model while keep the data decentralized.



McMahan et al. "Communication-efficient learning of deep networks from decentralized data." Artificial intelligence and statistics. PMLR, 2017.

Federated Learning

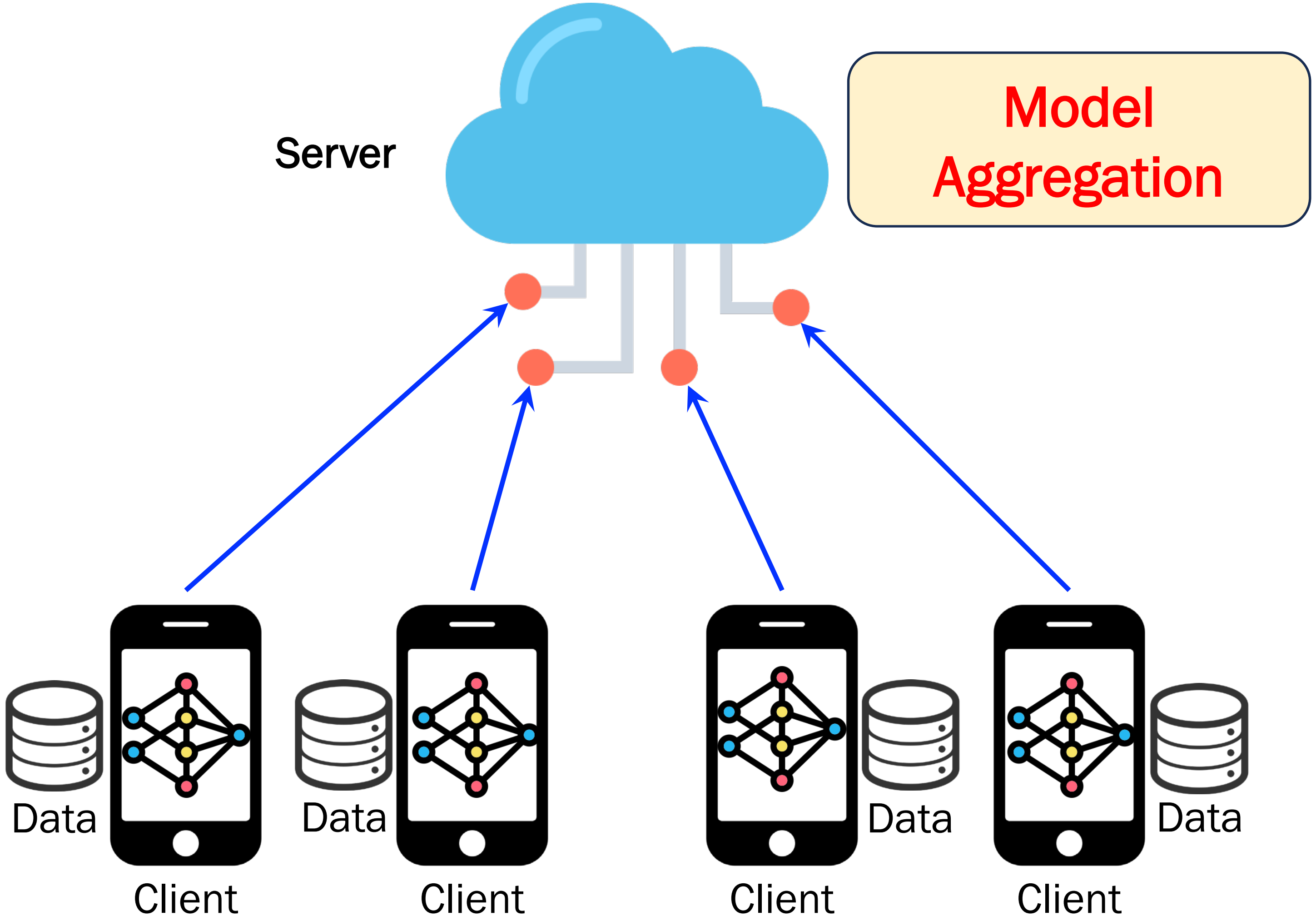
- Federated Learning (FL) aims to collaboratively train a machine learning (ML) model while keep the data decentralized.



McMahan et al. "Communication-efficient learning of deep networks from decentralized data." Artificial intelligence and statistics. PMLR, 2017.

Federated Learning

- Federated Learning (FL) aims to collaboratively train a machine learning (ML) model while keep the data decentralized.

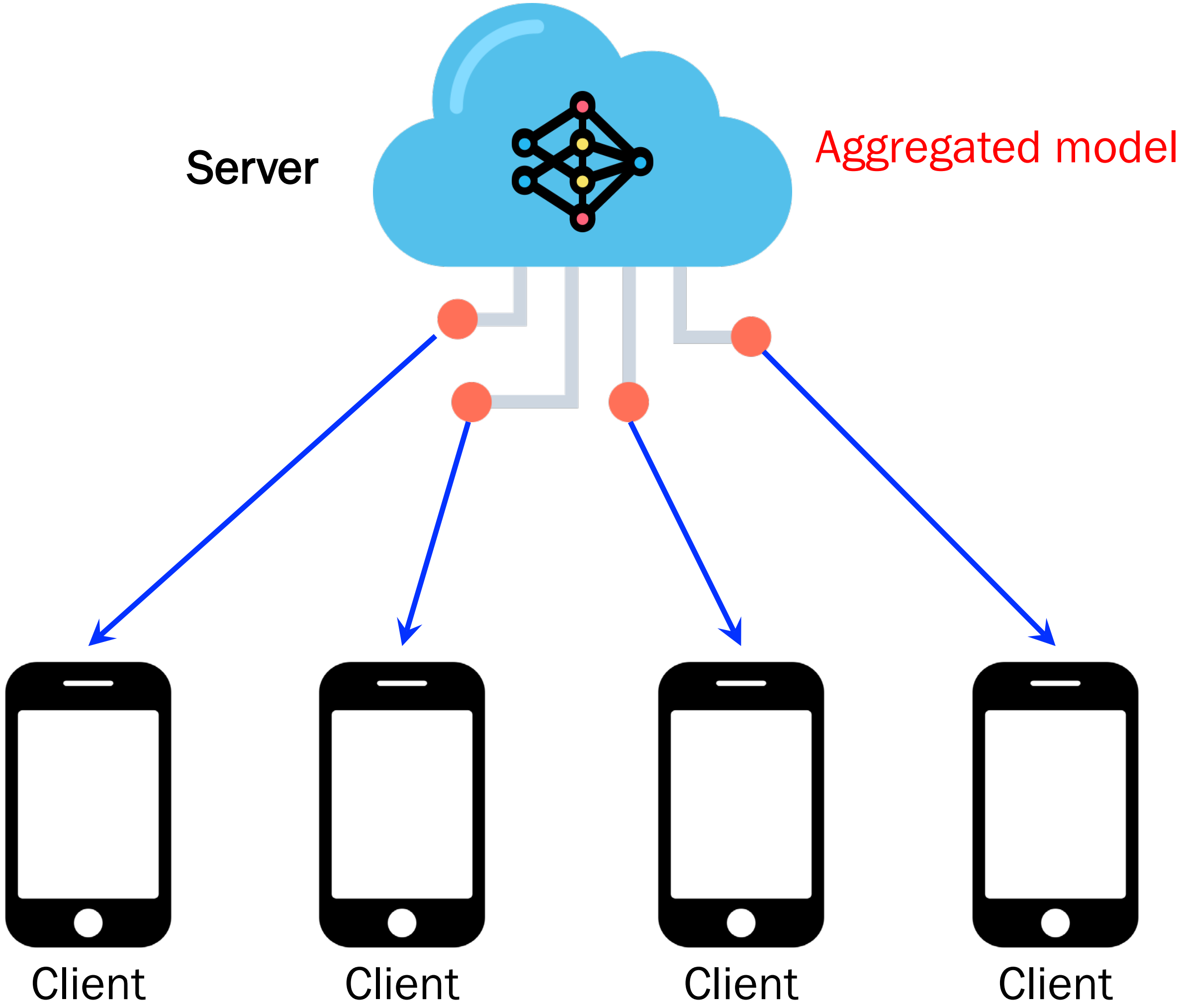


McMahan et al. "Communication-efficient learning of deep networks from decentralized data." Artificial intelligence and statistics. PMLR, 2017.

Federated Learning

- Federated Learning (FL) aims to collaboratively train a machine learning (ML) model while keep the data decentralized.

We would like the final aggregated model to be as good as the centralized solution (ideally), or at least better than what each client can learn on its own

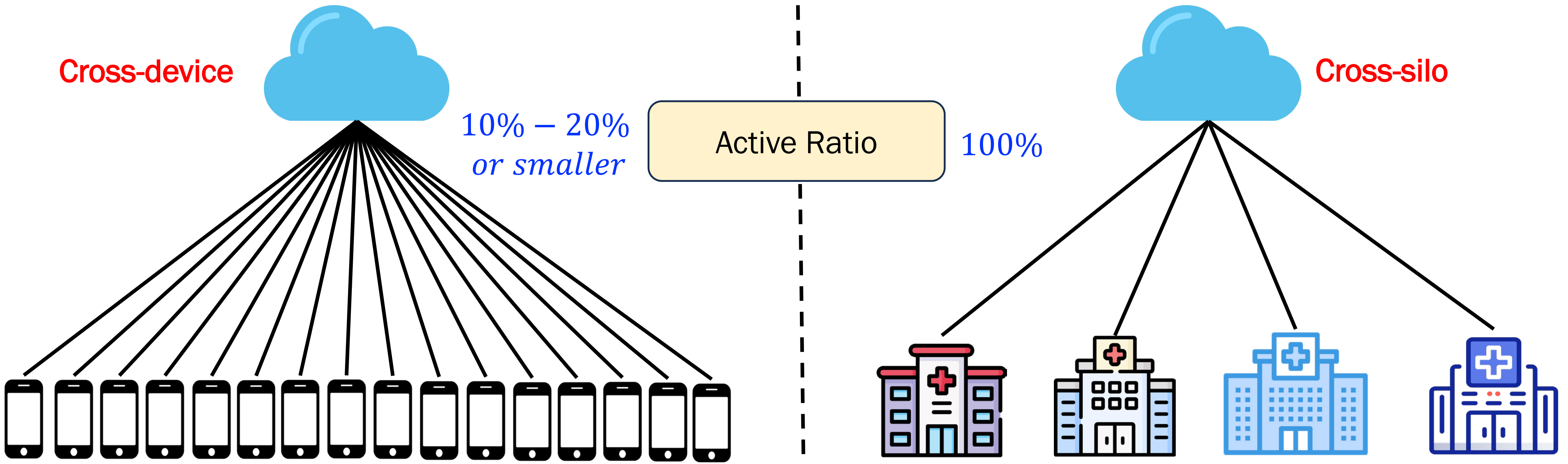


McMahan et al. "Communication-efficient learning of deep networks from decentralized data." Artificial intelligence and statistics. PMLR, 2017.

Taxonomy

- Cross-device vs. Cross-silo FL
  - Number of clients
- Vertical vs. Horizontal FL
  - Feature and sample
- Server-orchestrated vs. Fully-decentralized FL
  - Central server

Cross-device vs. Cross-silo Federated Learning



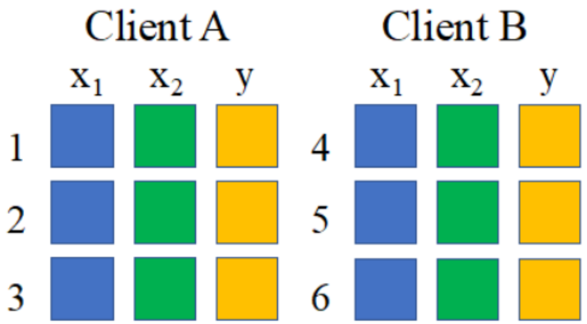
1. Massive number of clients (up to 10^{10})
2. Small dataset per client (could be size 1)
3. Limited availability and reliability
4. Some clients may be malicious

1. 2-100 clients
2. Medium to large dataset per client
3. Reliable clients, almost always available
4. Clients are typically honest

Horizontal vs. Vertical Federated Learning

- Horizontal FL:
  - Same feature space
  - Different sample space
  - Example: two banks may have different users from different regions, but their features can be same, e.g., job, age, gender, and credit score.

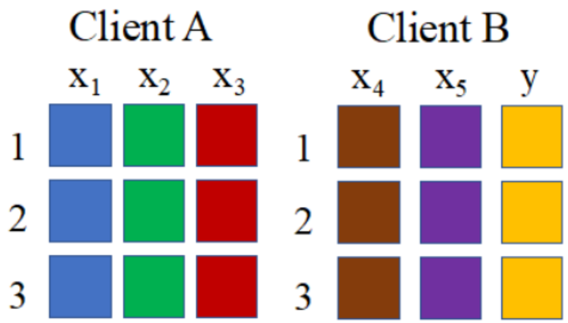
(a) Horizontal federated learning



Sample-based FL

- Vertical FL:
  - Different feature space
  - Same sample space
  - Example: a group of users have Facebook accounts and Amazon accounts. Facebook and Amazon have different features of the same group of users.

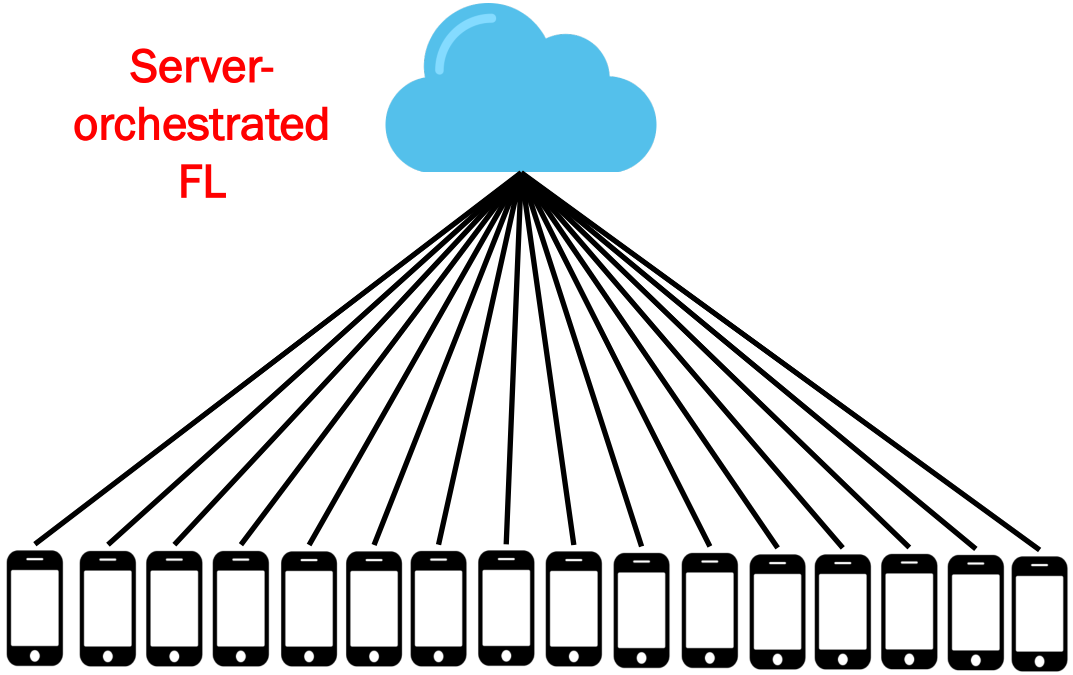
(b) Vertical federated learning



Feature-based FL

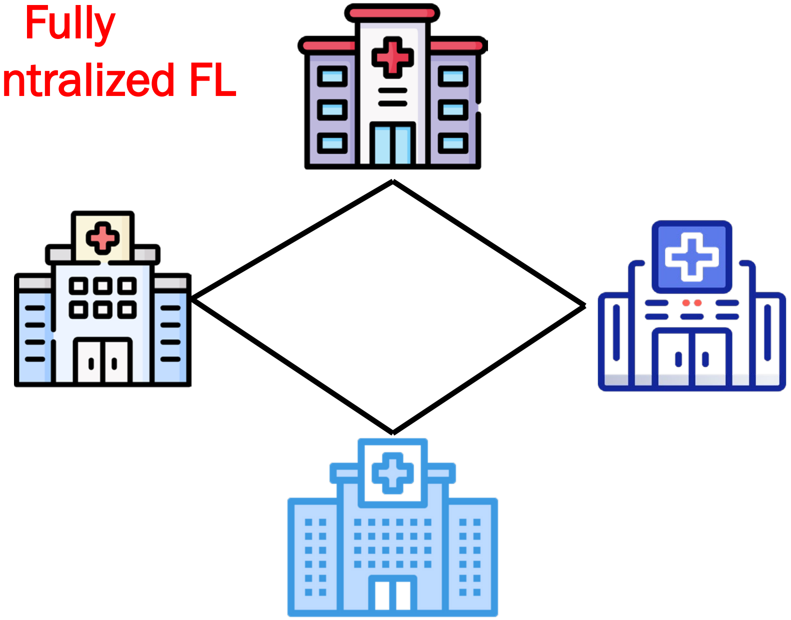
Yang et al. " Federated machine learning: Concept and applications." ACM Transactions on Intelligent Systems and Technology (TIST), 2019.

Server-orchestrated vs. Fully decentralized Federated Learning



Server-orchestrated FL

1. Server-client communication
2. Global coordination, global aggregation
3. Server is a single point of failure and may become a bottleneck

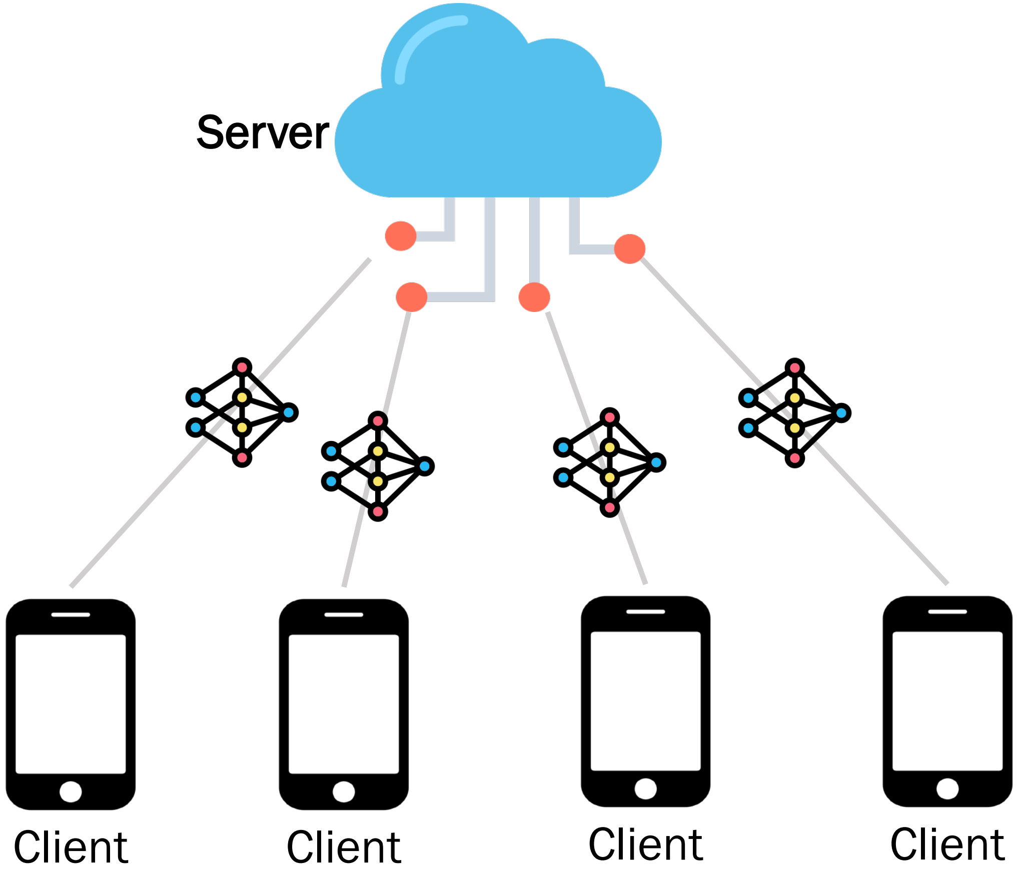


Fully decentralized FL

1. Client-to-client communication
2. No global coordination, local aggregation
3. Naturally scales to a large number of clients

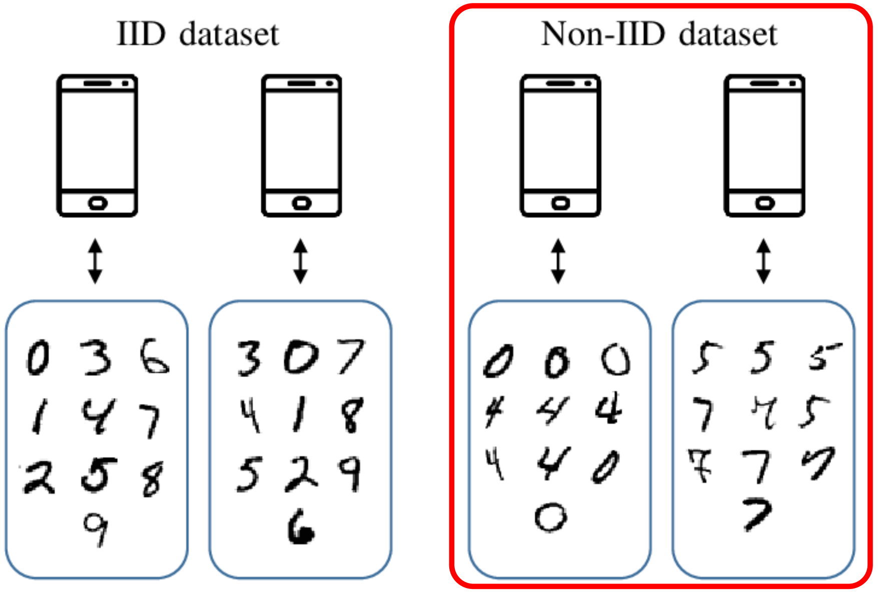
Core Challenges of Federated Learning

- Communication Efficiency
- Privacy Concerns
- Heterogeneity
  - Data/Statistical Heterogeneity
  - Model Heterogeneity
  - System Heterogeneity

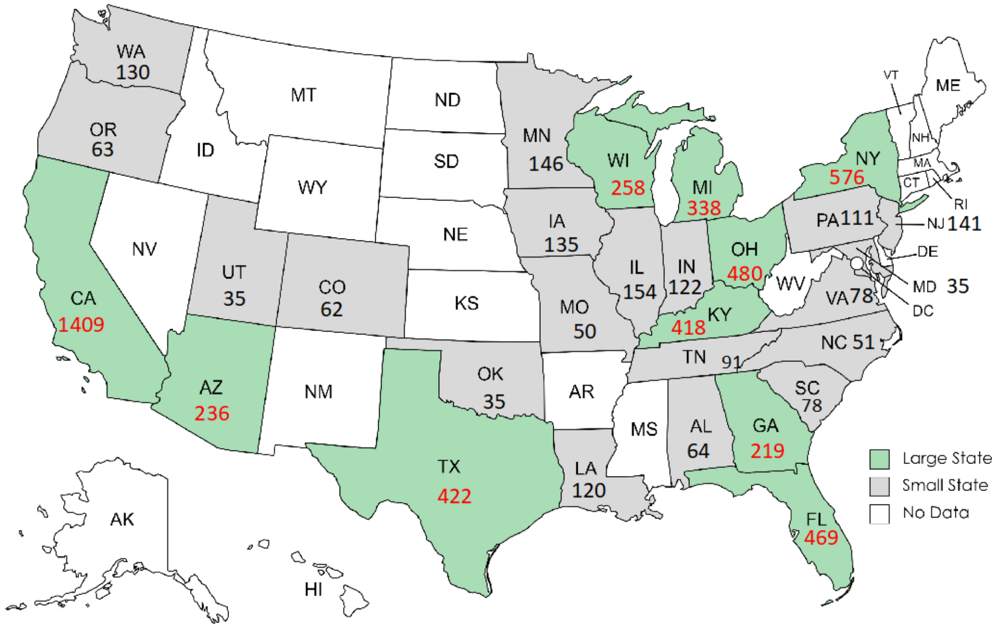


Data/Statistical Heterogeneity

Independent and Identically Distributed

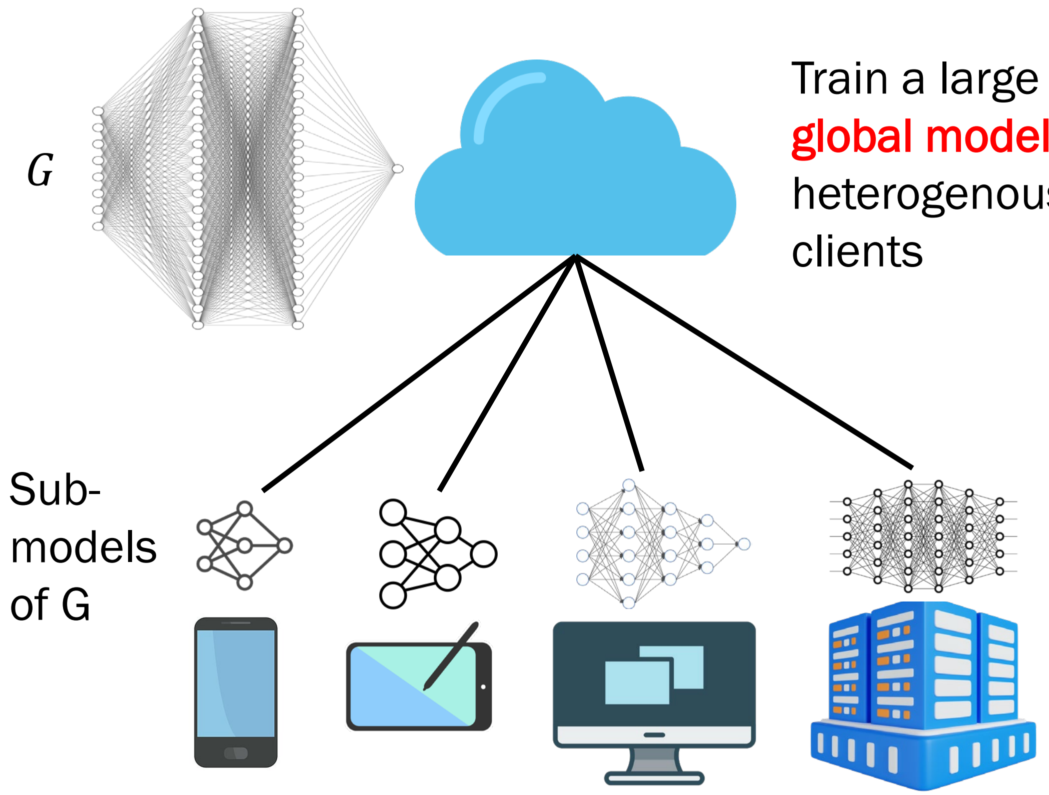


IID vs. non-IID for MNIST dataset



Patient geographical distribution across states in US

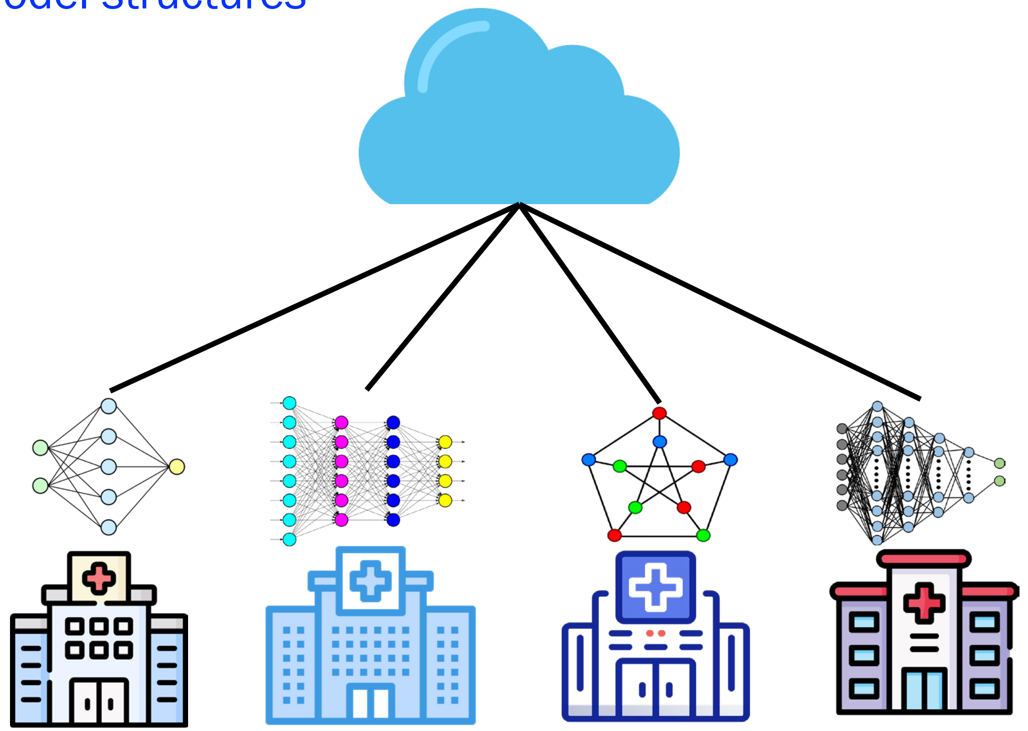
Model Heterogeneity



Train a large **global model** with heterogenous clients

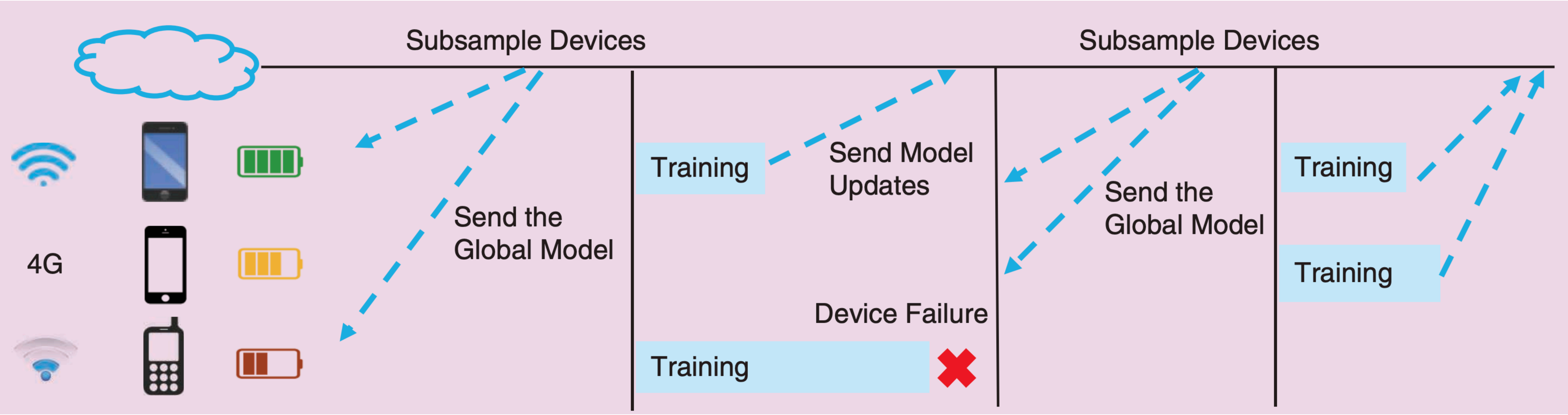
Sub-model training

Enhance the performance of each **client model** through collaborative learning *without modifying client model structures*



Heterogeneous model aggregation

System Heterogeneity



Devices may vary in terms of network connection, power, and hardware. Moreover, some of the devices may drop at any time during training.

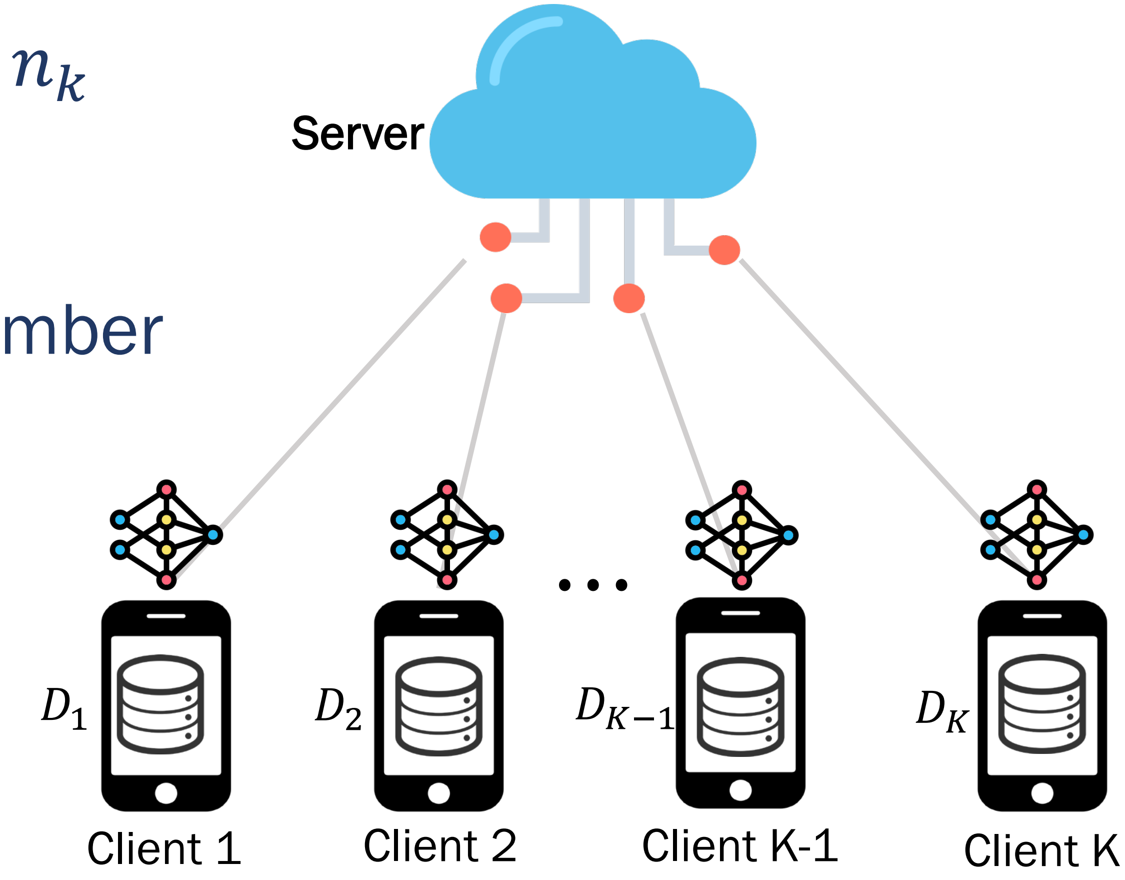
Li et al. " Federated Learning: Challenges, Methods, and Future Directions." IEEE signal processing magazine, 2020.

A Baseline Algorithm: FedAvg

- Each client k holds a dataset D_k of n_k samples
- Let $D = D_1 \cup \cdots \cup D_K$ be the join dataset and $n = \sum_k n_k$ the total number of samples
- Empirical risk minimization:

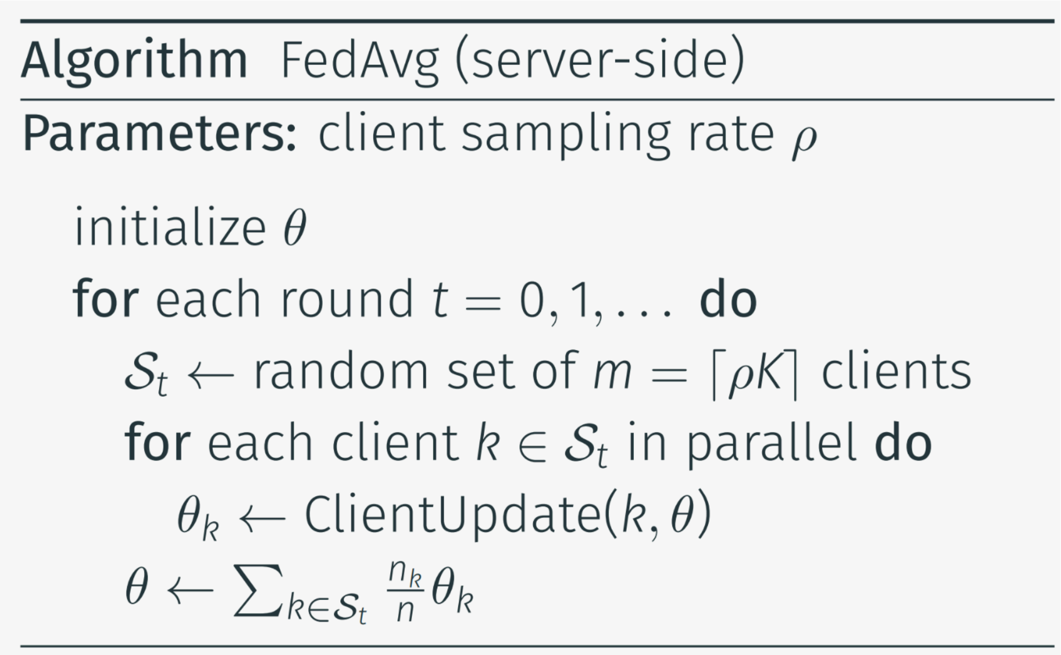
$$F(\theta; D) = \sum_{k=1}^{K} \frac{n_k}{n} F_k(\theta; D_k) \qquad F_k(\theta; D_k) = \sum_{d \in D_k} f(\theta; d)$$

$\theta \in \mathbb{R}^p$ are model parameters



McMahan et al. "Communication-efficient learning of deep networks from decentralized data." Artificial intelligence and statistics. PMLR, 2017.

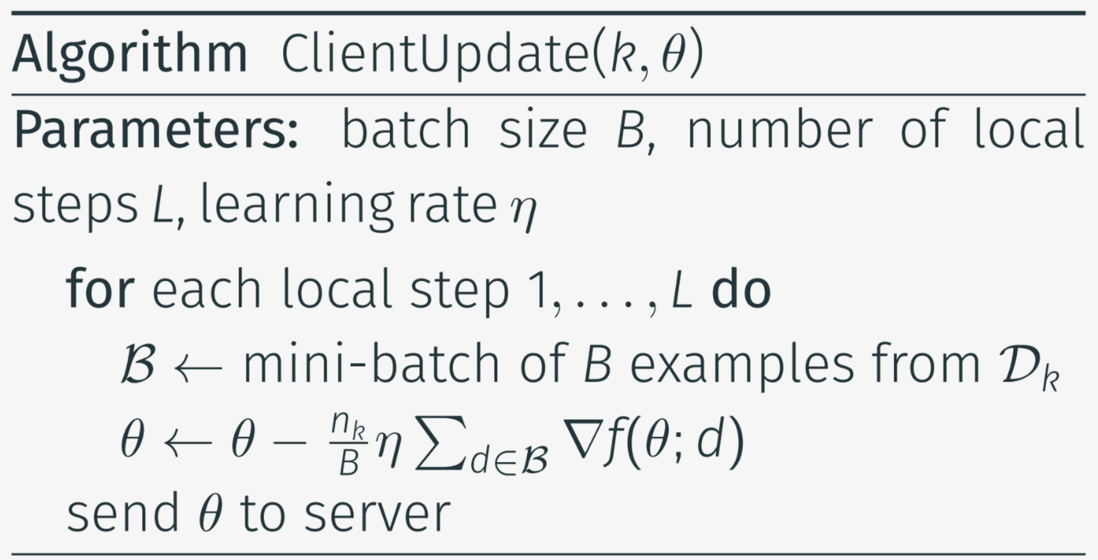
FedAvg



Algorithm FedAvg (server-side)

Parameters: client sampling rate ρ

```
initialize $\theta$
for each round $t = 0, 1, \ldots$ do
    $\mathcal{S}_t \leftarrow$ random set of $m = \lceil \rho K \rceil$ clients
    for each client $k \in \mathcal{S}_t$ in parallel do
        $\theta_k \leftarrow \text{ClientUpdate}(k, \theta)$
    $\theta \leftarrow \sum_{k \in \mathcal{S}_t} \frac{n_k}{n} \theta_k$
```



Algorithm ClientUpdate(k, θ)

Parameters: batch size B, number of local steps L, learning rate η

```
for each local step $1, \ldots, L$ do
    $\mathcal{B} \leftarrow$ mini-batch of $B$ examples from $\mathcal{D}_k$
    $\theta \leftarrow \theta - \frac{n_k}{B} \eta \sum_{d \in \mathcal{B}} \nabla f(\theta; d)$
send $\theta$ to server
```

- For $L = 1$ and $\rho = 1$, it is equivalent to classic parallel SGD: updates are aggregated, and the model synchronized at each step
- For $L > 1$: each client performs multiple local SGD steps before communicating

McMahan et al. "Communication-efficient learning of deep networks from decentralized data." Artificial intelligence and statistics. PMLR, 2017.

Part 2

- Part 1: Federated Learning Introduction
- Part 2: Data/Statistical Heterogeneity
- Part 3: Model Heterogeneity
- Part 4: System Heterogeneity
- Part 5: Conclusion and Future Work

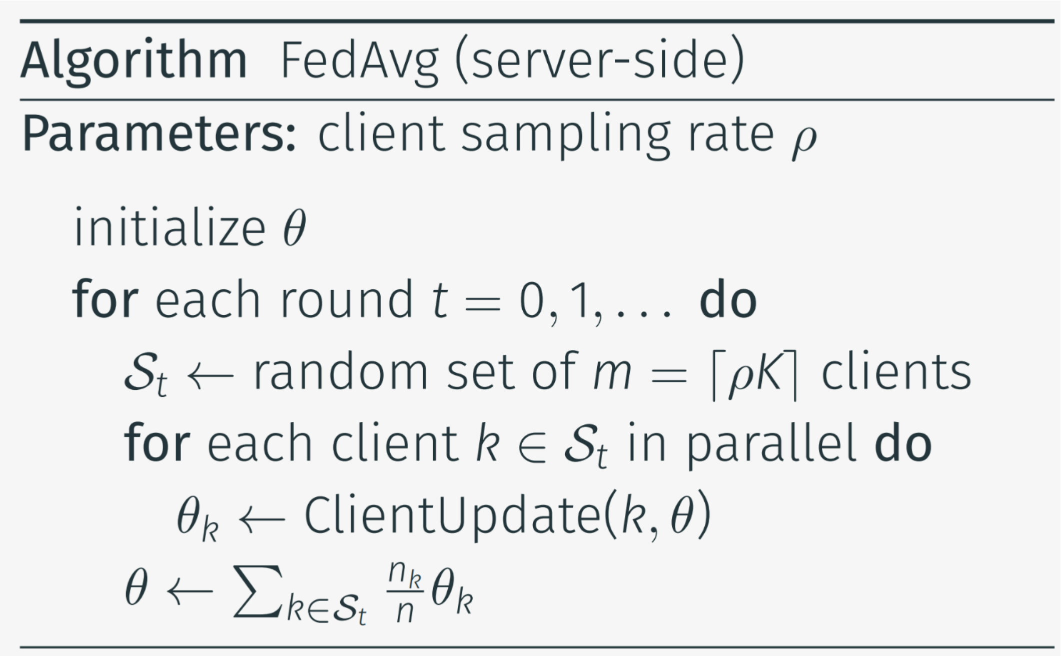
Approaches

- Regularization
  - FedProx
- Clustering
- Data Augmentation
- Multimodal Disentanglement

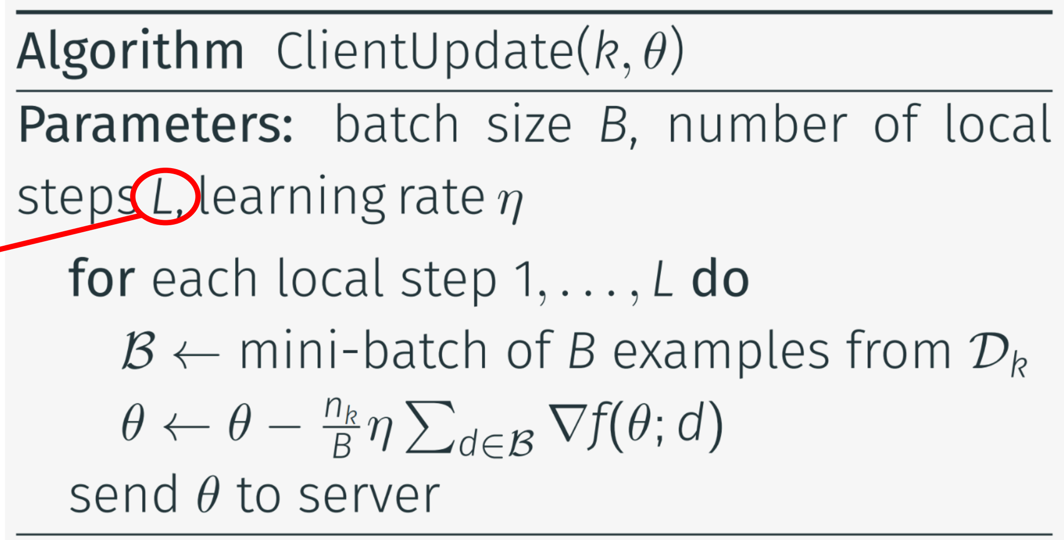
FedProx

- Drawbacks of FedAvg
  - Different devices in federated networks often have **different resource constraints** in terms of the computing hardware, network connections, and battery levels
  - Unrealistic to force each device to perform a uniform amount of work

Running the same number of local epochs for all clients



```
Algorithm FedAvg (server-side)
Parameters: client sampling rate $\rho$
  initialize $\theta$
  for each round $t = 0, 1, \dots$ do
    $\mathcal{S}_t \leftarrow$ random set of $m = \lceil \rho K \rceil$ clients
    for each client $k \in \mathcal{S}_t$ in parallel do
      $\theta_k \leftarrow$ ClientUpdate$(k, \theta)$
    $\theta \leftarrow \sum_{k \in \mathcal{S}_t} \frac{n_k}{n} \theta_k$
```



```
Algorithm ClientUpdate$(k, \theta)$
Parameters: batch size $B$, number of local steps $L$, learning rate $\eta$
  for each local step $1, \dots, L$ do
    $\mathcal{B} \leftarrow$ mini-batch of $B$ examples from $\mathcal{D}_k$
    $\theta \leftarrow \theta - \frac{n_k}{B} \eta \sum_{d \in \mathcal{B}} \nabla f(\theta; d)$
  send $\theta$ to server
```

PennState

Li et al. "Federated optimization in heterogeneous networks." Proceedings of Machine learning and systems 2 (2020): 429-450.

FedProx

- Add a proximal term to the local subproblem to effectively limit the impact of variable local updates

$$\min_{w} h_k(w;\ w^t) = F_k(w) + \frac{\mu}{2}\|w - \boxed{w^t}\|^2$$

The aggregated model from the server at time t.

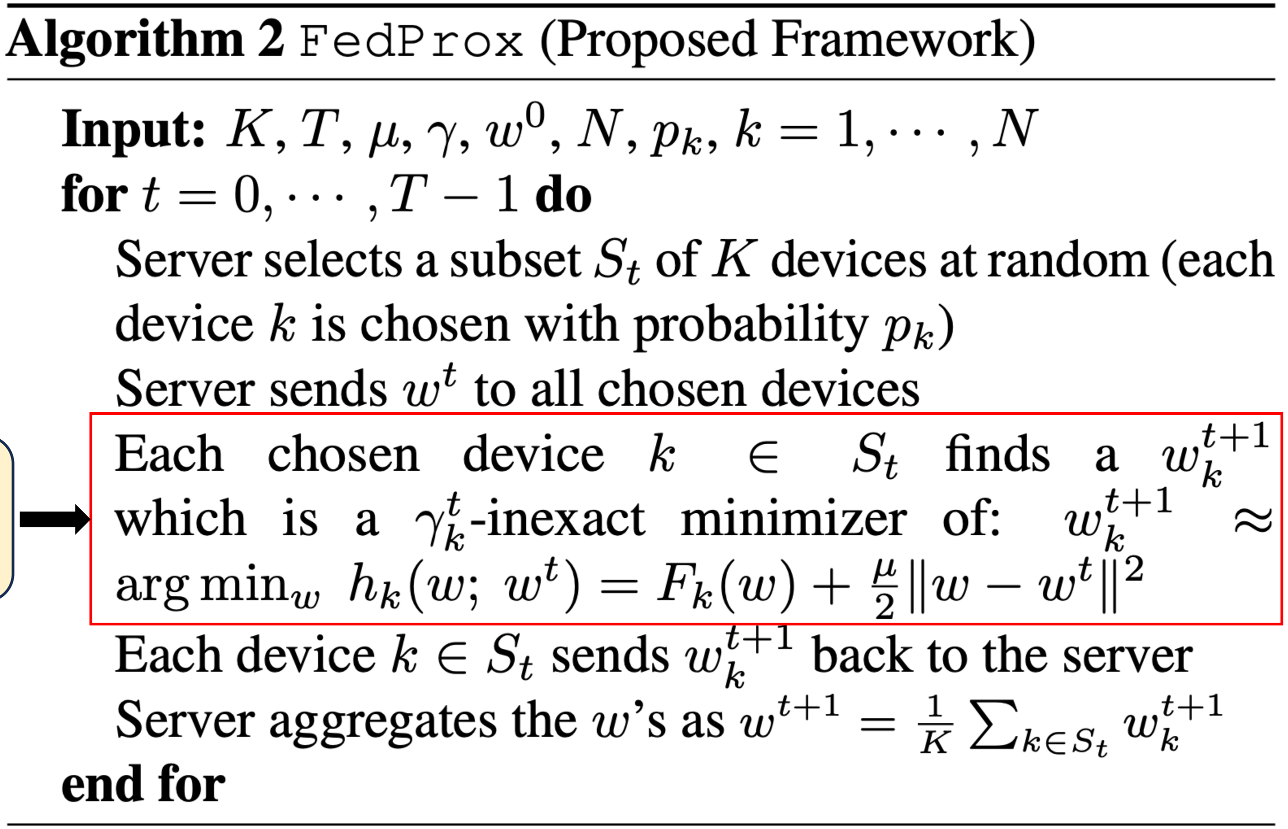
- It addresses the issue of statistical heterogeneity by restricting the local updates to be closer to the initial (global) model without any need to manually set the number of local epochs.
- It allows for safely incorporating variable amounts of local work resulting from systems heterogeneity.

PennState

Li et al. "Federated optimization in heterogeneous networks." Proceedings of Machine learning and systems 2 (2020): 429-450.

FedProx

$$\min_{w} h_k(w;\ w^t) = F_k(w) + \frac{\mu}{2}\|w - w^t\|^2$$



```
Algorithm 2 FedProx (Proposed Framework)
Input: $K, T, \mu, \gamma, w^0, N, p_k, k = 1, \cdots, N$
for $t = 0, \cdots, T-1$ do
    Server selects a subset $S_t$ of $K$ devices at random (each device $k$ is chosen with probability $p_k$)
    Server sends $w^t$ to all chosen devices
    Each chosen device $k \in S_t$ finds a $w_k^{t+1}$ which is a $\gamma_k^t$-inexact minimizer of: $w_k^{t+1} \approx \arg\min_w \ h_k(w;\ w^t) = F_k(w) + \frac{\mu}{2}\|w - w^t\|^2$
    Each device $k \in S_t$ sends $w_k^{t+1}$ back to the server
    Server aggregates the $w$'s as $w^{t+1} = \frac{1}{K}\sum_{k \in S_t} w_k^{t+1}$
end for
```

No number of local steps L

K: Selected clients

T: Communication round

μ, γ: Hyperparameters

w^0: Initialized model

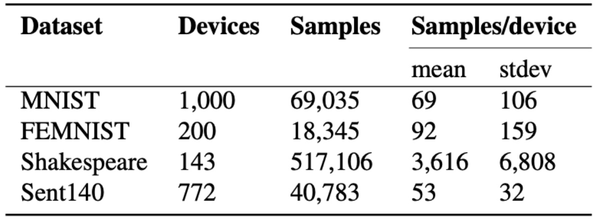
N: # of clients

$p_k = \frac{n_k}{n}$

Li et al. "Federated optimization in heterogeneous networks." Proceedings of Machine learning and systems 2 (2020): 429-450.

FedProx

- Results



Dataset	Devices	Samples	Samples/device	
			mean	stdev
MNIST	1,000	69,035	69	106
FEMNIST	200	18,345	92	159
Shakespeare	143	517,106	3,616	6,808
Sent140	772	40,783	53	32

$$\min_{w} h_k(w;\ w^t) = F_k(w) + \frac{\mu}{2}\|w - w^t\|^2$$

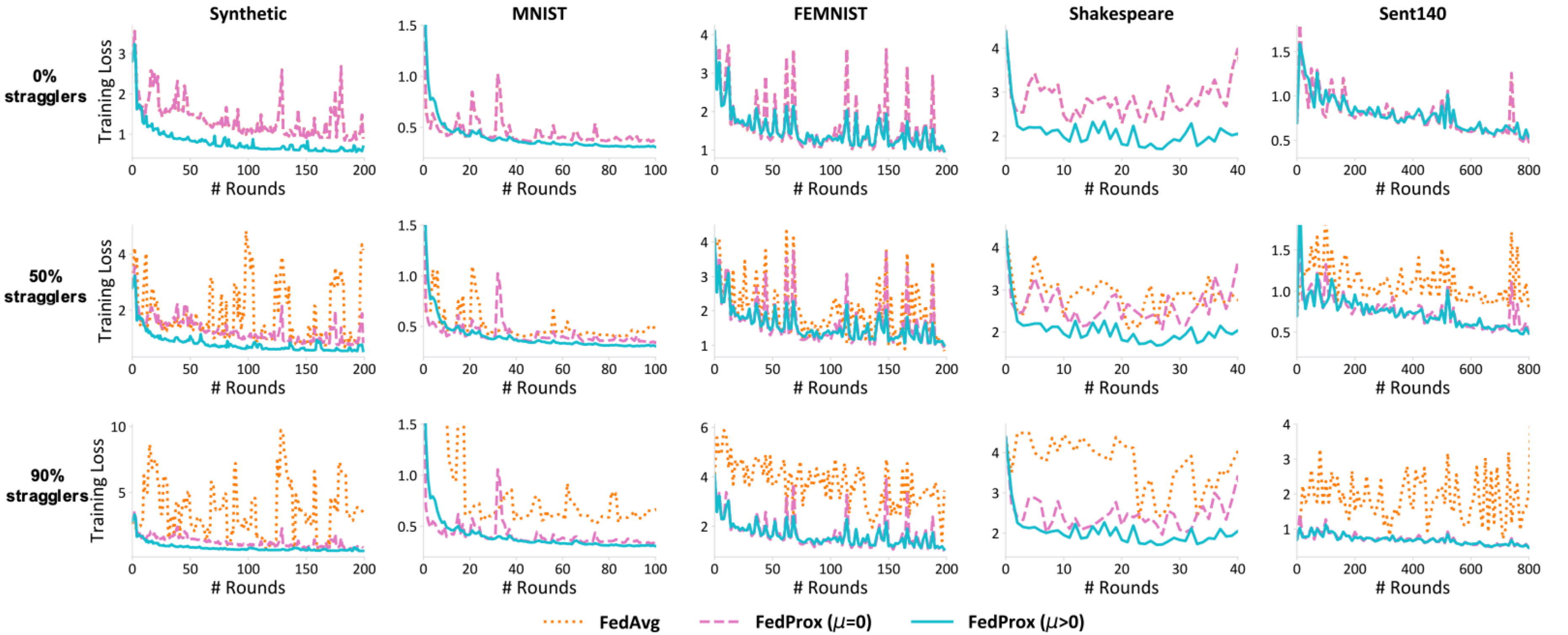


Figure 1. FedProx results in significant convergence improvements relative to FedAvg in heterogeneous networks. We simulate different levels of systems heterogeneity by forcing 0%, 50%, and 90% devices to be the stragglers (dropped by FedAvg). (1) Comparing FedAvg and FedProx ($\mu = 0$), we see that allowing for variable amounts of work to be performed can help convergence in the presence of systems heterogeneity. (2) Comparing FedProx ($\mu = 0$) with FedProx ($\mu > 0$), we show the benefits of our added proximal term. FedProx with $\mu > 0$ leads to more stable convergence and enables otherwise divergent methods to converge, both in the presence of systems heterogeneity (50% and 90% stragglers) and without systems heterogeneity (0% stragglers). Note that FedProx with $\mu = 0$ and without systems heterogeneity (no stragglers) corresponds to FedAvg. We also report testing accuracy in Figure 7, Appendix C.3.2, and show that FedProx improves the test accuracy on all datasets.

Li et al. "Federated optimization in heterogeneous networks." Proceedings of Machine learning and systems 2 (2020): 429-450.

FedProx

- Results

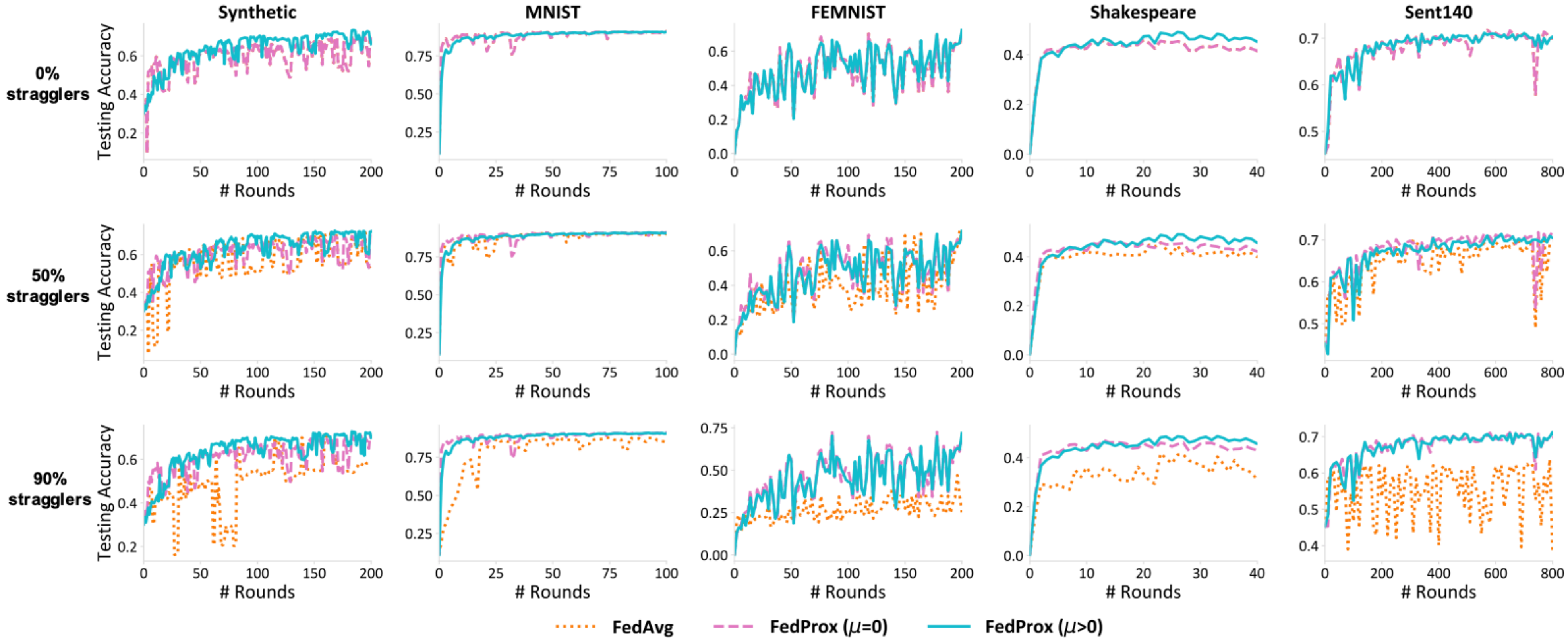


Figure 7. The testing accuracy of the experiments in Figure 1. FedProx achieves on average 22% improvement in terms of testing accuracy in highly heterogeneous settings (90% stragglers).

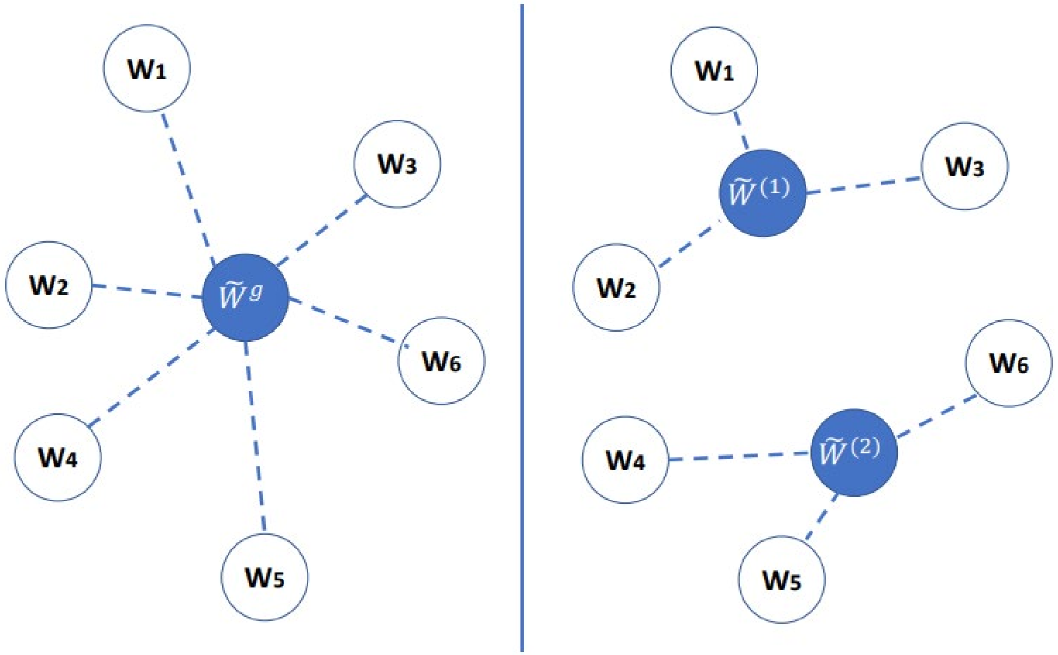
Li et al. "Federated optimization in heterogeneous networks." Proceedings of Machine learning and systems 2 (2020): 429-450.

Approaches

- Regularization
  - FedProx
- Clustering
  - FedSEM
- Data Augmentation
- Multimodal Disentanglement

FedSEM

- Existing FL approaches
  - Update a single global model to capture the shared knowledge of all users by aggregating their gradients, regardless of the discrepancy between their data distributions.
- Solution
  - A mixture of multiple global models could capture the heterogeneity across various clients if assigning the client to different global models (i.e., centers) in FL.



Long et al. "Multi-center federated learning: clients clustering for better personalization." World Wide Web 26.1 (2023): 481-500.

FedSEM

- The multi-center FL problem can be formulated as joint optimization problem:

The parameters of the aggregated model for cluster-k.

$$\min_{\{W_i\},\{r_i^{(k)}\},\{\tilde{W}^{(k)}\}} \sum_{i=1}^{m} \alpha_i L_s(\mathcal{M}_i, \mathcal{D}_i, W_i) + \frac{\lambda}{m} \sum_{k=1}^{K} \sum_{i=1}^{m} r_i^{(k)} \, \mathrm{Dist}(W_i, \tilde{W}^{(k)}),$$

Multi-center assignment at the server end.

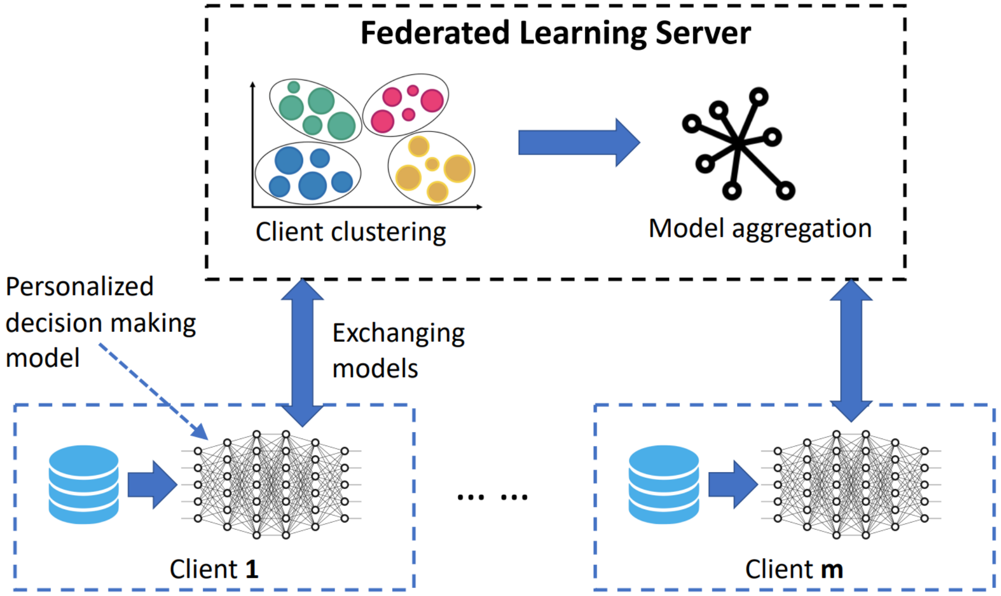
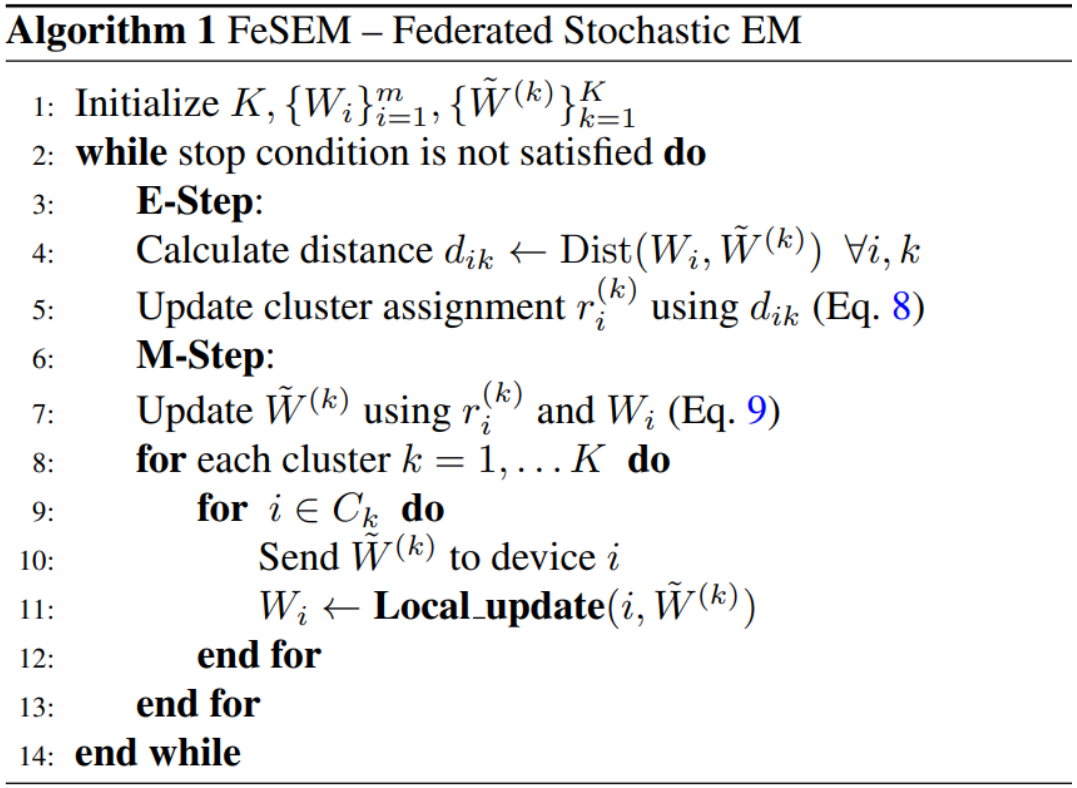


Fig. 1: Overall framework of multi-center Federated Learning.

- On each node-i: optimize W_i , while fixing others;
- On the server: optimize $r_i^{(k)}$, $\widetilde{W}_i^k$ while fixing all the local models.

Long et al. "Multi-center federated learning: clients clustering for better personalization." World Wide Web 26.1 (2023): 481-500.

FedSEM



Algorithm 1 FeSEM – Federated Stochastic EM

```
Initialize $K, \{W_i\}_{i=1}^m, \{\tilde{W}^{(k)}\}_{k=1}^K$
while stop condition is not satisfied do
    E-Step:
    Calculate distance $d_{ik} \leftarrow \text{Dist}(W_i, \tilde{W}^{(k)}) \;\; \forall i, k$
    Update cluster assignment $r_i^{(k)}$ using $d_{ik}$ (Eq. 8)
    M-Step:
    Update $\tilde{W}^{(k)}$ using $r_i^{(k)}$ and $W_i$ (Eq. 9)
    for each cluster $k = 1, \dots K$ do
        for $i \in C_k$ do
            Send $\tilde{W}^{(k)}$ to device $i$
            $W_i \leftarrow$ Local_update$(i, \tilde{W}^{(k)})$
        end for
    end for
end while
```

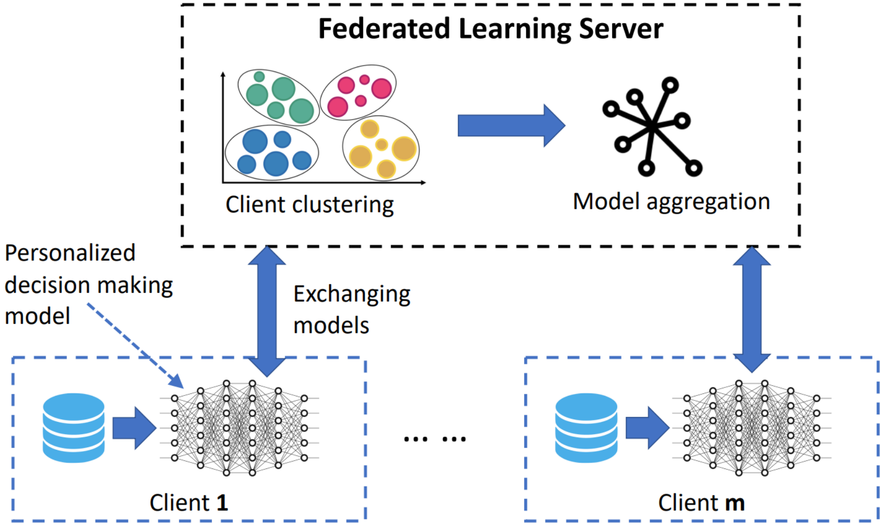
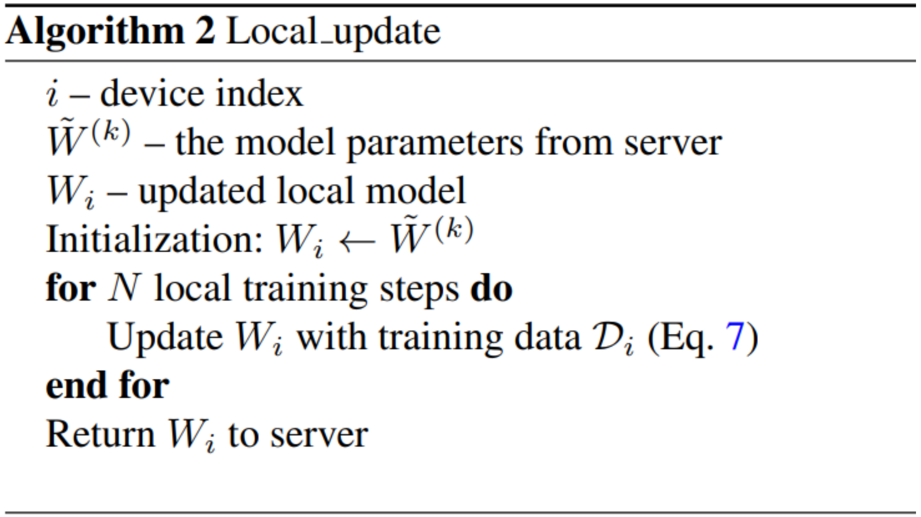


Fig. 1: Overall framework of multi-center Federated Learning.

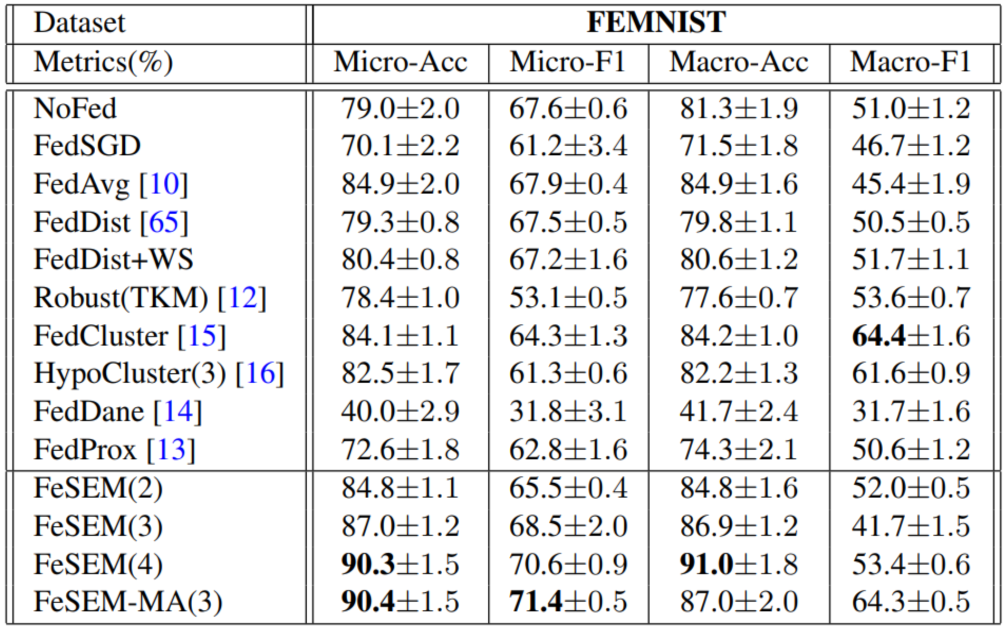


Algorithm 2 Local_update

```
$i$ – device index
$\tilde{W}^{(k)}$ – the model parameters from server
$W_i$ – updated local model
Initialization: $W_i \leftarrow \tilde{W}^{(k)}$
for $N$ local training steps do
    Update $W_i$ with training data $\mathcal{D}_i$ (Eq. 7)
end for
Return $W_i$ to server
```

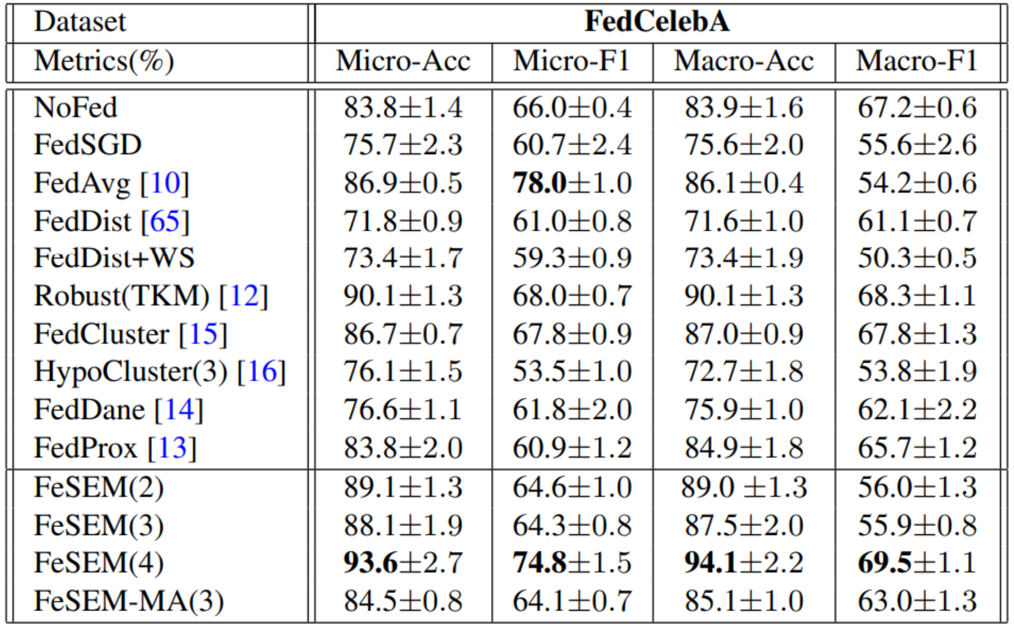
Long et al. "Multi-center federated learning: clients clustering for better personalization." World Wide Web 26.1 (2023): 481-500.

FedSEM



Dataset	FEMNIST			
Metrics(%)	Micro-Acc	Micro-F1	Macro-Acc	Macro-F1
NoFed	79.0±2.0	67.6±0.6	81.3±1.9	51.0±1.2
FedSGD	70.1±2.2	61.2±3.4	71.5±1.8	46.7±1.2
FedAvg [10]	84.9±2.0	67.9±0.4	84.9±1.6	45.4±1.9
FedDist [65]	79.3±0.8	67.5±0.5	79.8±1.1	50.5±0.5
FedDist+WS	80.4±0.8	67.2±1.6	80.6±1.2	51.7±1.1
Robust(TKM) [12]	78.4±1.0	53.1±0.5	77.6±0.7	53.6±0.7
FedCluster [15]	84.1±1.1	64.3±1.3	84.2±1.0	**64.4**±1.6
HypoCluster(3) [16]	82.5±1.7	61.3±0.6	82.2±1.3	61.6±0.9
FedDane [14]	40.0±2.9	31.8±3.1	41.7±2.4	31.7±1.6
FedProx [13]	72.6±1.8	62.8±1.6	74.3±2.1	50.6±1.2
FeSEM(2)	84.8±1.1	65.5±0.4	84.8±1.6	52.0±0.5
FeSEM(3)	87.0±1.2	68.5±2.0	86.9±1.2	41.7±1.5
FeSEM(4)	**90.3**±1.5	70.6±0.9	**91.0**±1.8	53.4±0.6
FeSEM-MA(3)	**90.4**±1.5	**71.4**±0.5	87.0±2.0	64.3±0.5

Table 2: Comparison of our proposed FeSEM(K) algorithm with the baselines on FEMNIST. Note the number in parenthesis following "FeSEM" denotes the number of clusters, K.



Dataset	FedCelebA			
Metrics(%)	Micro-Acc	Micro-F1	Macro-Acc	Macro-F1
NoFed	83.8±1.4	66.0±0.4	83.9±1.6	67.2±0.6
FedSGD	75.7±2.3	60.7±2.4	75.6±2.0	55.6±2.6
FedAvg [10]	86.9±0.5	**78.0**±1.0	86.1±0.4	54.2±0.6
FedDist [65]	71.8±0.9	61.0±0.8	71.6±1.0	61.1±0.7
FedDist+WS	73.4±1.7	59.3±0.9	73.4±1.9	50.3±0.5
Robust(TKM) [12]	90.1±1.3	68.0±0.7	90.1±1.3	68.3±1.1
FedCluster [15]	86.7±0.7	67.8±0.9	87.0±0.9	67.8±1.3
HypoCluster(3) [16]	76.1±1.5	53.5±1.0	72.7±1.8	53.8±1.9
FedDane [14]	76.6±1.1	61.8±2.0	75.9±1.0	62.1±2.2
FedProx [13]	83.8±2.0	60.9±1.2	84.9±1.8	65.7±1.2
FeSEM(2)	89.1±1.3	64.6±1.0	89.0 ±1.3	56.0±1.3
FeSEM(3)	88.1±1.9	64.3±0.8	87.5±2.0	55.9±0.8
FeSEM(4)	**93.6**±2.7	**74.8**±1.5	**94.1**±2.2	**69.5**±1.1
FeSEM-MA(3)	84.5±0.8	64.1±0.7	85.1±1.0	63.0±1.3

Table 3: Comparison of our proposed FeSEM(K) algorithm with the baselines on FedCelebA. Note the number in parenthesis following "FeSEM" denotes the number of clusters, K.

PennState

Long et al. "Multi-center federated learning: clients clustering for better personalization." World Wide Web 26.1 (2023): 481-500.

FedSEM

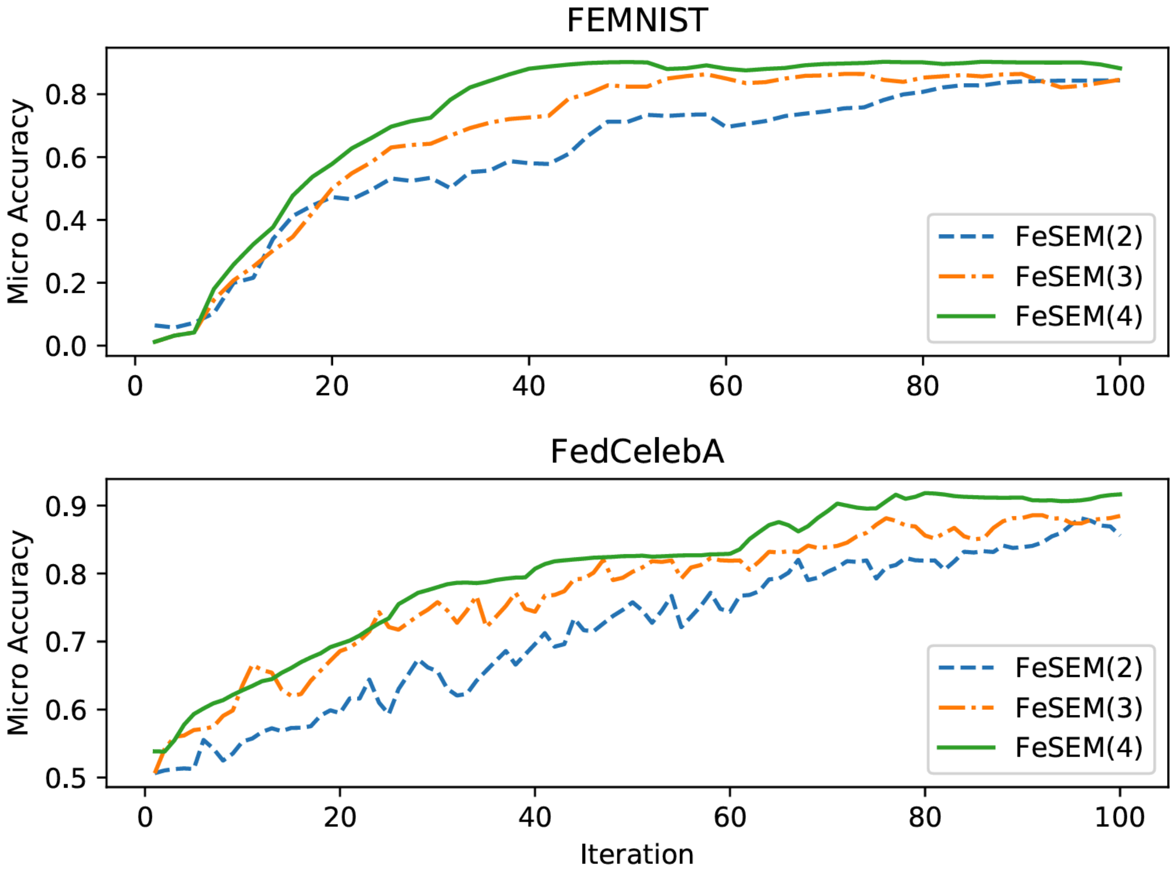


Fig. 3: Convergence analysis for the proposed FeSEM with different cluster number (in parenthesis) in terms of micro-accuracy.

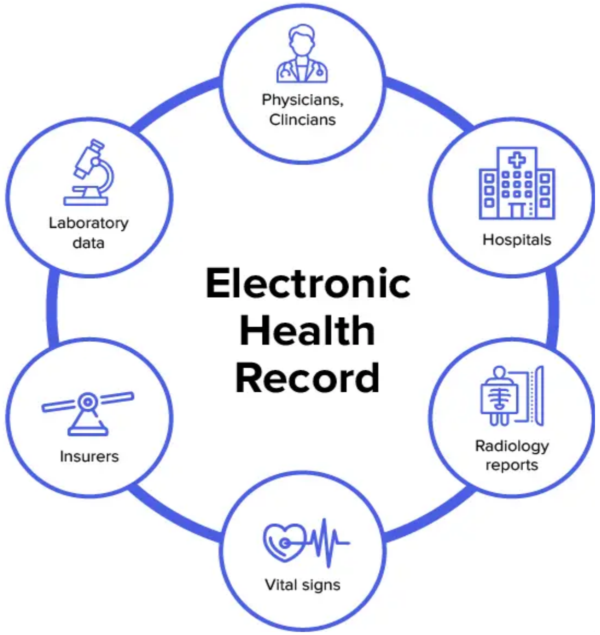
Long et al. "Multi-center federated learning: clients clustering for better personalization." World Wide Web 26.1 (2023): 481-500.

Approaches

- Regularization
  - FedProx
- Clustering
  - FedSEM
- Data Augmentation
  - FedCovid
- Multimodal Disentanglement

FedCovid

- Predicting Covid-19 vaccination with federated learning using electronic health records (EHR)
  - Each state in US is a client.
- Challenges
  - EHR data are heterogeneous.

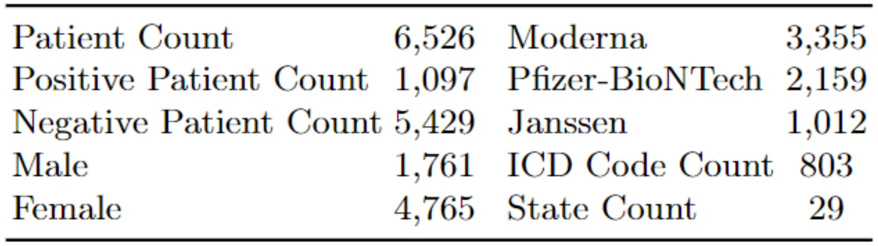


Wang et al. “Towards federated covid-19 vaccine side effect prediction.” Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD). 2022.

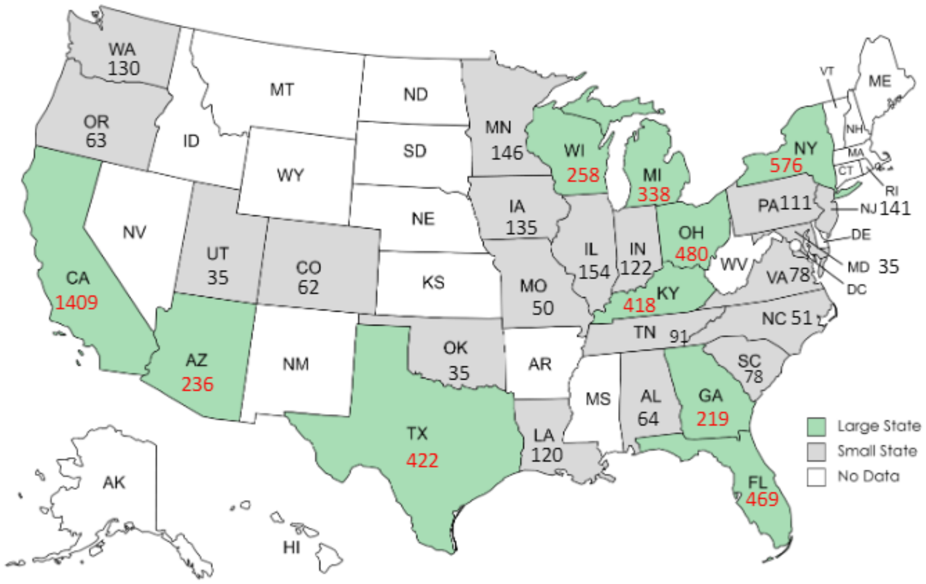
FedCovid

- Predicting Covid-19 vaccination with federated learning using electronic health records (EHR)
    - Each state in US is a client
- Challenges
    - EHR data are heterogeneous.
    - The size of EHR data stored for each client is unequal.

Table 1: Data statistics of the extracted EHR dataset.



Patient Count	6,526	Moderna	3,355
Positive Patient Count	1,097	Pfizer-BioNTech	2,159
Negative Patient Count	5,429	Janssen	1,012
Male	1,761	ICD Code Count	803
Female	4,765	State Count	29

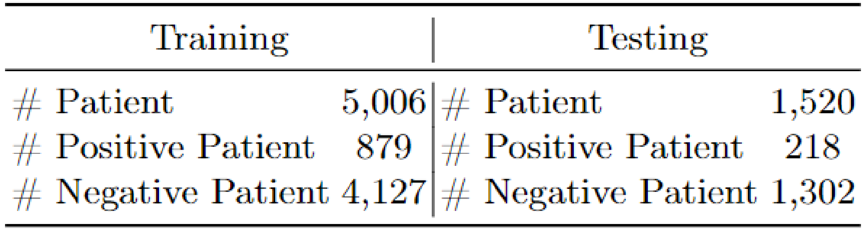


Wang et al. "Towards federated covid-19 vaccine side effect prediction." Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD). 2022.

FedCovid

- Data Imbalanced Heterogeneity

Table 2: Training and testing data statistics.



Training		Testing	
# Patient	5,006	# Patient	1,520
# Positive Patient	879	# Positive Patient	218
# Negative Patient	4,127	# Negative Patient	1,302

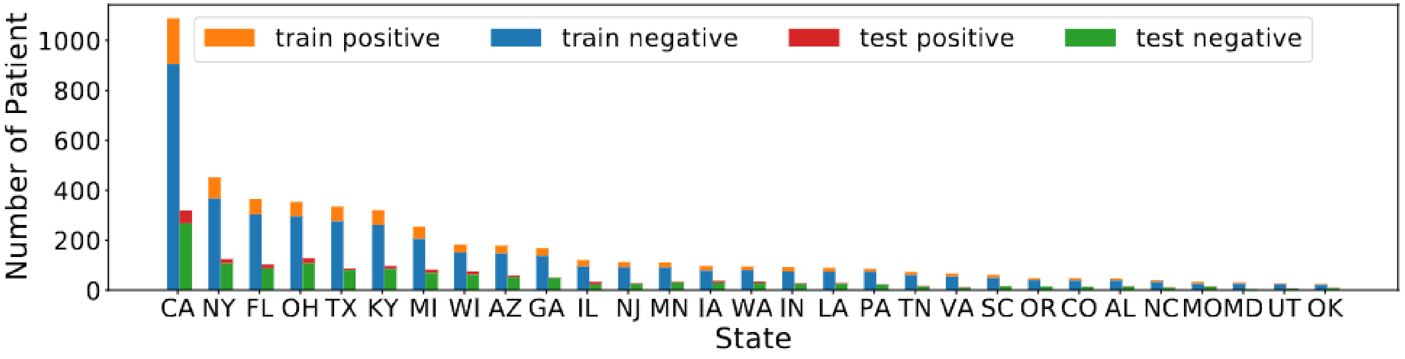
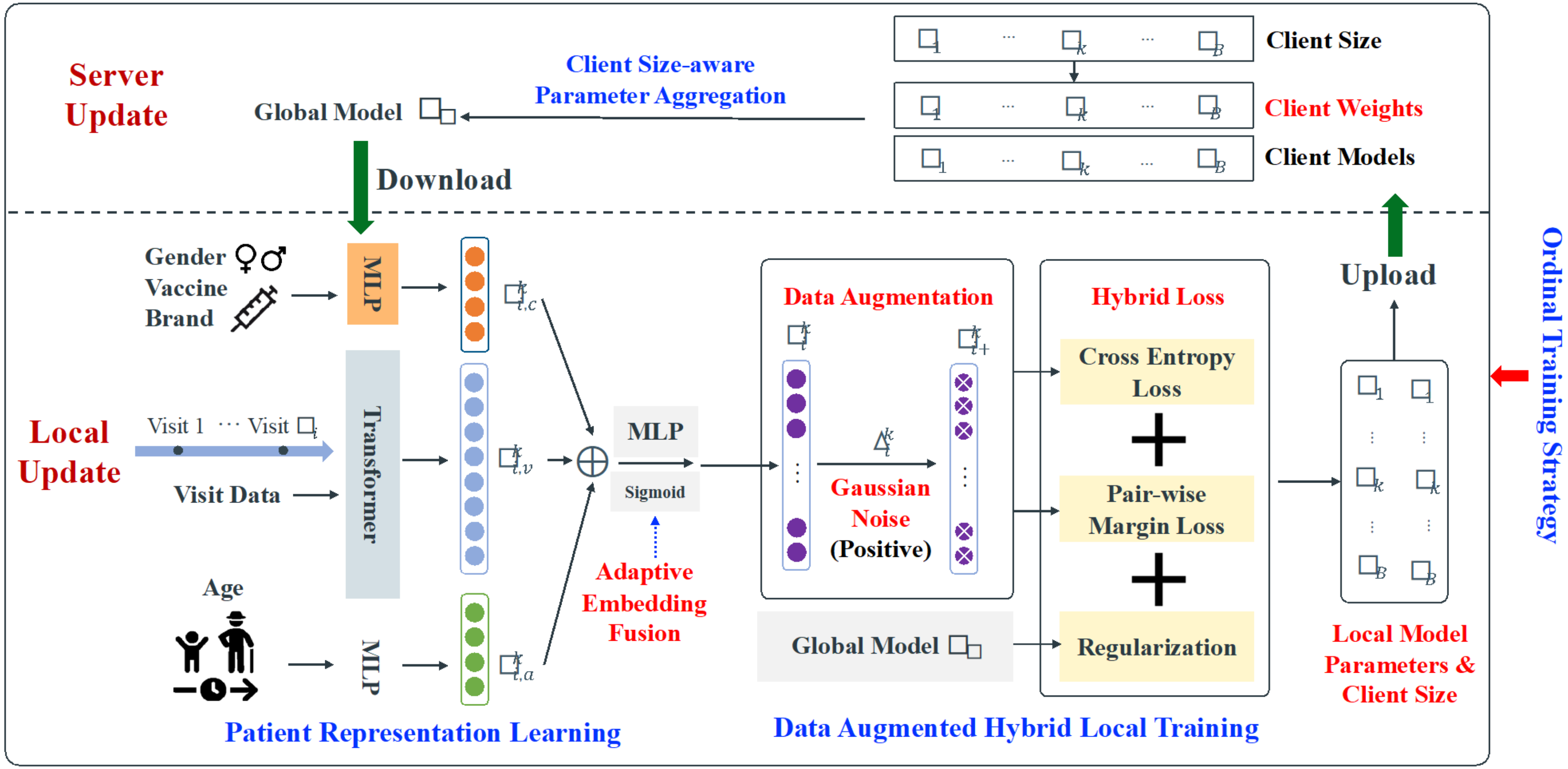


Fig. 2: Training and test data label ratio for each state.

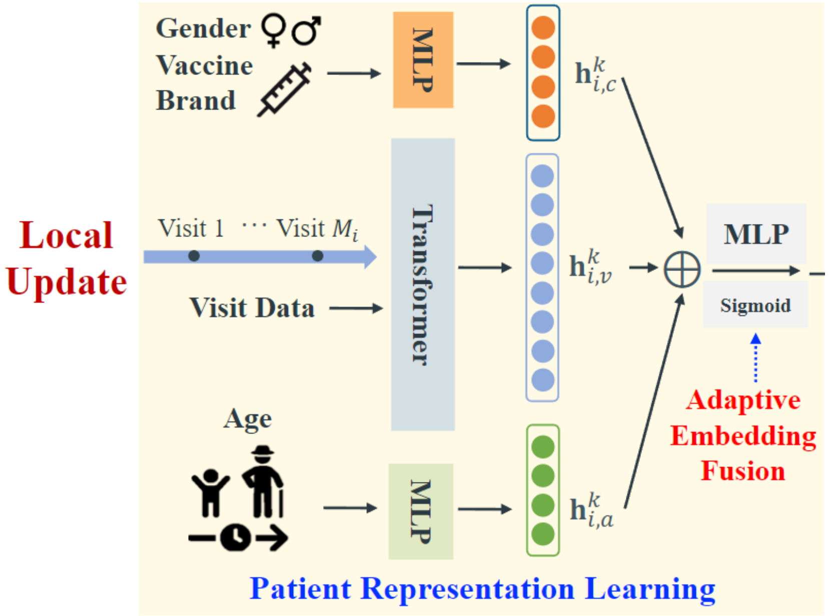
Wang et al. “Towards federated covid-19 vaccine side effect prediction.” Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD). 2022.

FedCovid



Wang et al. “Towards federated covid-19 vaccine side effect prediction.” Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD). 2022.

FedCovid



- Embedding Numerical and Categorical Features

$$\mathbf{h}^k_{i,a} = \mathrm{MLP}_a(a^k_i); \quad \mathbf{h}^k_{i,c} = \mathrm{MLP}_c(g^k_i, b^k_i).$$

Age information

Brand information

- Embedding Sequential Visit Data

$$\mathbf{h}^k_{i,v} = \mathcal{M}_b\left(V^k_i\right)$$

Visit information

- Adaptive Embedding Fusion

$$\mathbf{h}^{k\prime}_i = \mathbf{W}^k_i \mathbf{h}^k_i, \qquad \mathbf{h}^k_i = [\mathbf{h}^k_{i,a}, \mathbf{h}^k_{i,c}, \mathbf{h}^k_{i,v}]$$

where $\mathbf{W}^k_i$ is a learnable weight matrix. We then learn a weight for each element in $\mathbf{h}^{k\prime}_i$ via a Sigmoid function, i.e.,

$$\phi^k_i = \mathrm{sigmoid}(\mathbf{h}^{k\prime}_i).$$

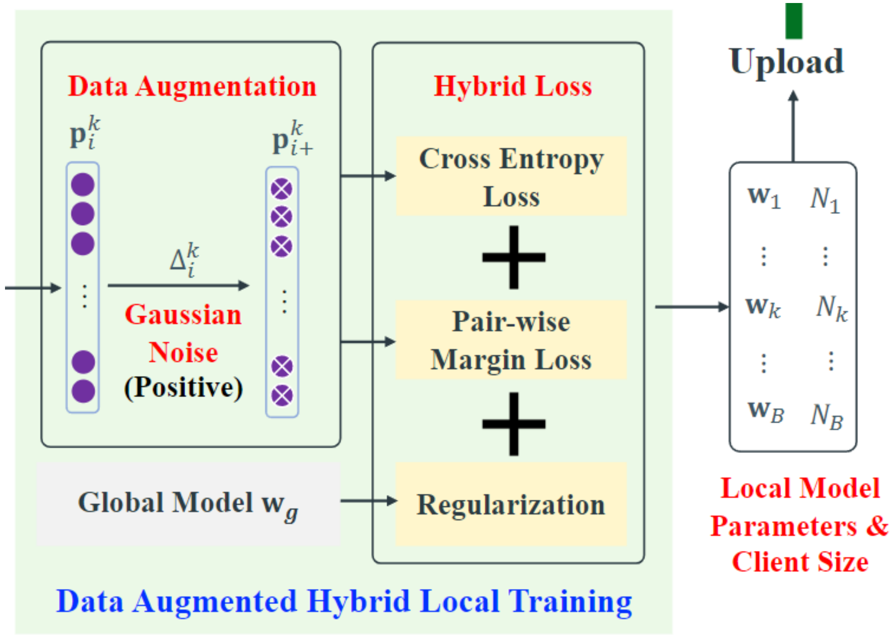
Finally, the element-wise multiplication $\circ$ is used to generate the patient representation as follows:

$$\mathbf{p}^k_i = \phi^k_i \mathbf{h}^{k\prime}_i.$$

Wang et al. “Towards federated covid-19 vaccine side effect prediction.” Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD). 2022.

FedCovid

- EHR Data Augmentation
- Hybrid Local Training



$$\mathcal{L}_c^k = \frac{1}{N_k}\mathrm{CE}(f(\mathbf{P}^k), \mathbf{y}^k) + \frac{\lambda_c}{N_k^+}\mathrm{CE}(f(\hat{\mathbf{P}}_+^k), \mathbf{y}_+^k),$$

Representation matrix of the augmented positive data

Pair-wise margin loss:

$$\mathcal{L}_m^k = \frac{1}{N_k + N_k^+}\sum_{i=1}^{N_k+N_k^+}\max(d(\tilde{\mathbf{p}}_i^k, \bar{\mathbf{p}}_{j+}^k) - d(\tilde{\mathbf{p}}_i^k, \mathbf{p}_{j'-}^k) + \delta, 0),$$

Final hybrid loss:

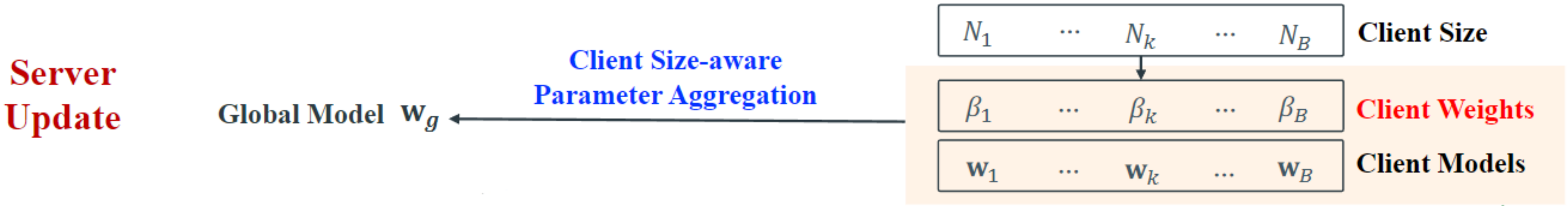
$$\mathcal{L}_k = \mathcal{L}_c^k + \lambda_m \mathcal{L}_m^k + \frac{\lambda_w}{N_w}\left\|\mathbf{w}_k - \mathbf{w}_g\right\|^2,$$

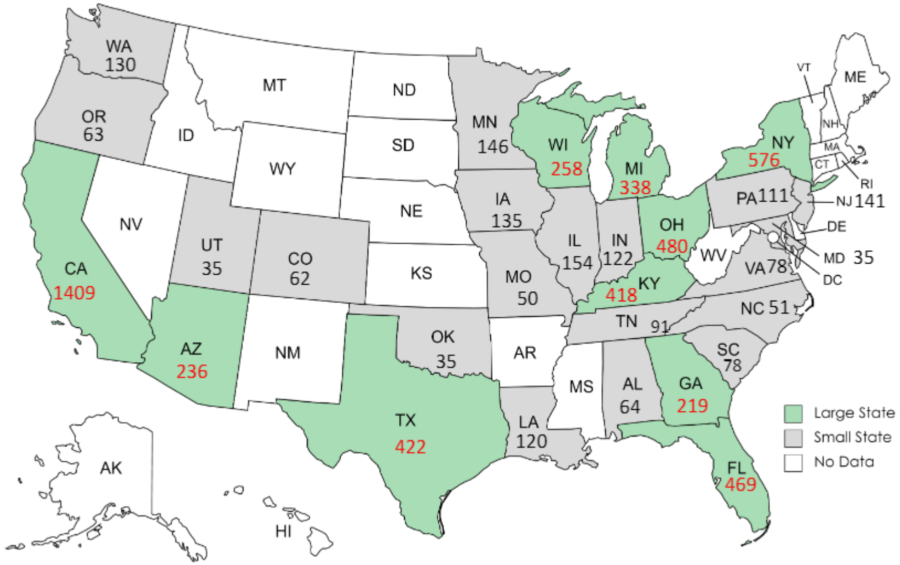
Number of model parameters

PennState

Wang et al. “Towards federated covid-19 vaccine side effect prediction.” Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD). 2022.

FedCovid


Server Update
Global Model $\mathbf{w}_g$
Client Size-aware Parameter Aggregation
N_1 ··· N_k ··· N_B
Client Size
β_1 ··· β_k ··· β_B
Client Weights
$\mathbf{w}_1$ ··· $\mathbf{w}_k$ ··· $\mathbf{w}_B$
Client Models



- Server Update: Client Size-aware Aggregation

$$\mathbf{w}_g = \frac{1}{B}\sum_{k=1}^{B} \beta_k * \mathbf{w}_k. \qquad \beta_k = \frac{\log(N_k)}{\sum_{i=1}^{B}\log(N_i)}.$$

Number of model parameters

- Ordinal Training Strategy:
  - First train clients with larger size and then train small clients
  - For the small client training, we lower the number of training epochs and learning rate.

We try to lower the negative effect caused by the smaller clients.

PennState

Wang et al. “Towards federated covid-19 vaccine side effect prediction.” Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD). 2022.

FedCovid

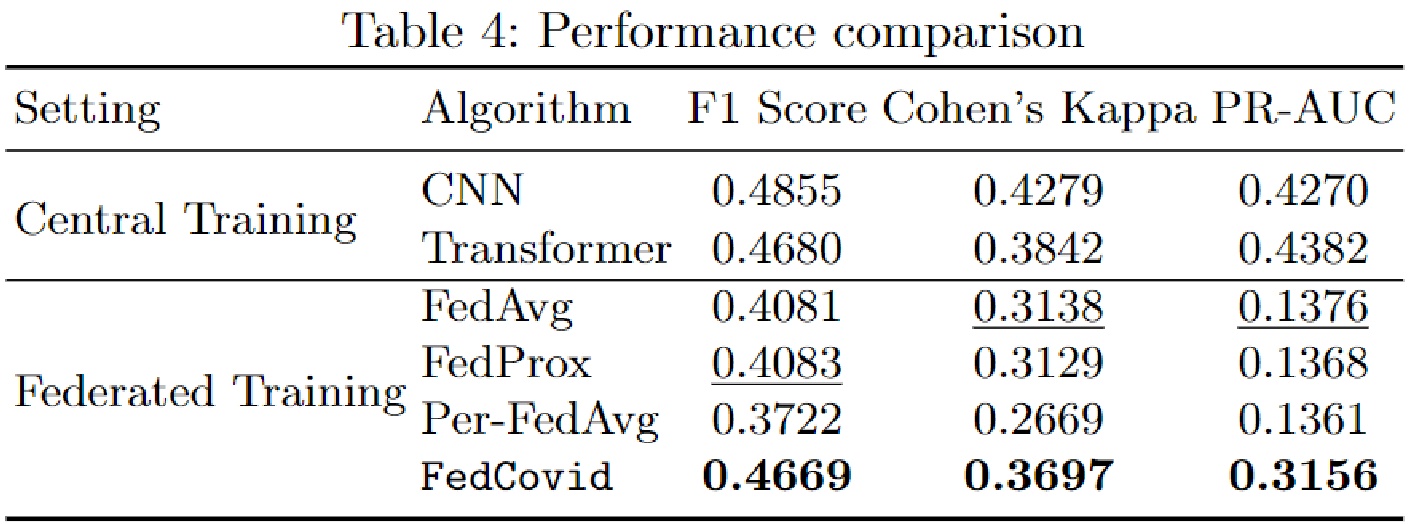


Table 4: Performance comparison

Setting	Algorithm	F1 Score	Cohen's Kappa	PR-AUC
Central Training	CNN	0.4855	0.4279	0.4270
	Transformer	0.4680	0.3842	0.4382
Federated Training	FedAvg	0.4081	0.3138	0.1376
	FedProx	0.4083	0.3129	0.1368
	Per-FedAvg	0.3722	0.2669	0.1361
	FedCovid	**0.4669**	**0.3697**	**0.3156**

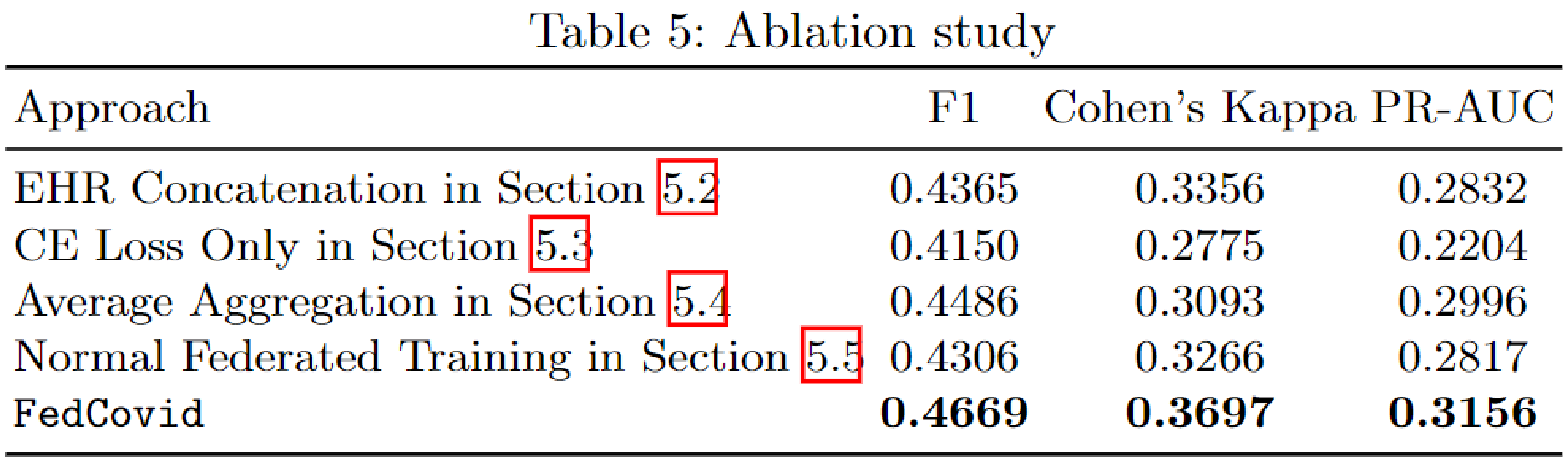


Table 5: Ablation study

Approach	F1	Cohen's Kappa	PR-AUC
EHR Concatenation in Section 5.2	0.4365	0.3356	0.2832
CE Loss Only in Section 5.3	0.4150	0.2775	0.2204
Average Aggregation in Section 5.4	0.4486	0.3093	0.2996
Normal Federated Training in Section 5.5	0.4306	0.3266	0.2817
FedCovid	**0.4669**	**0.3697**	**0.3156**

PennState

Wang et al. "Towards federated covid-19 vaccine side effect prediction." Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD). 2022.

Approaches

- Regularization
  - FedProx
- Clustering
  - FedSEM
- Data Augmentation
  - FedCovid
- Multimodal Disentanglement
  - Harmony

Harmony

- Federated multi-modal sensing systems

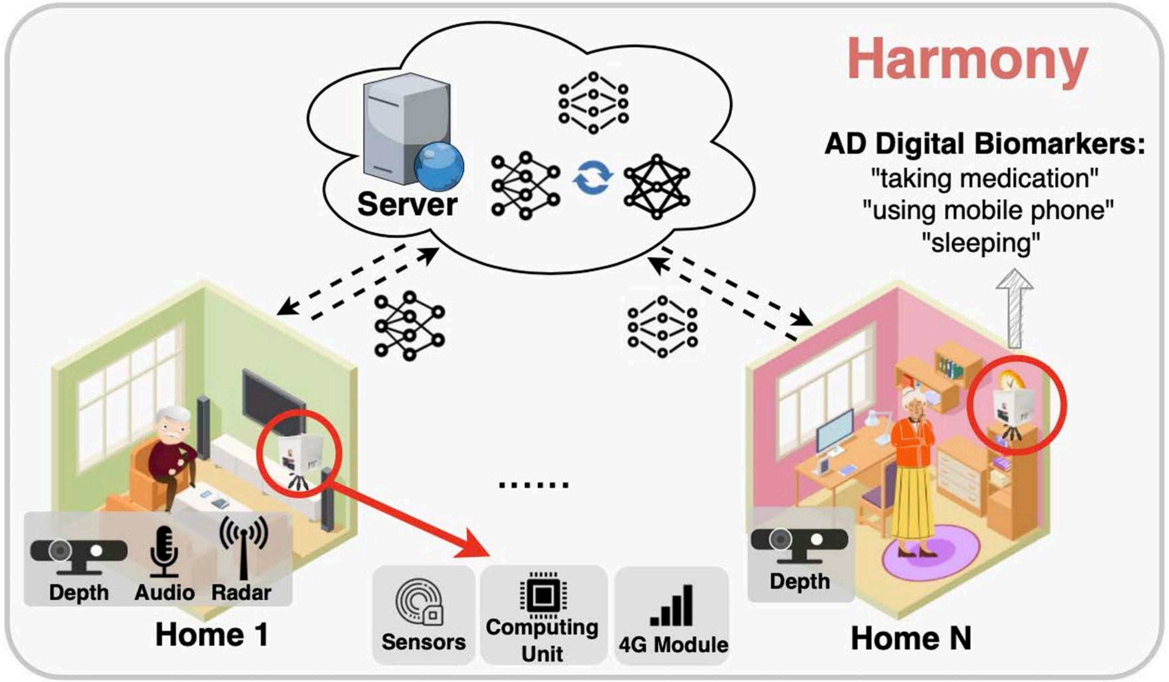


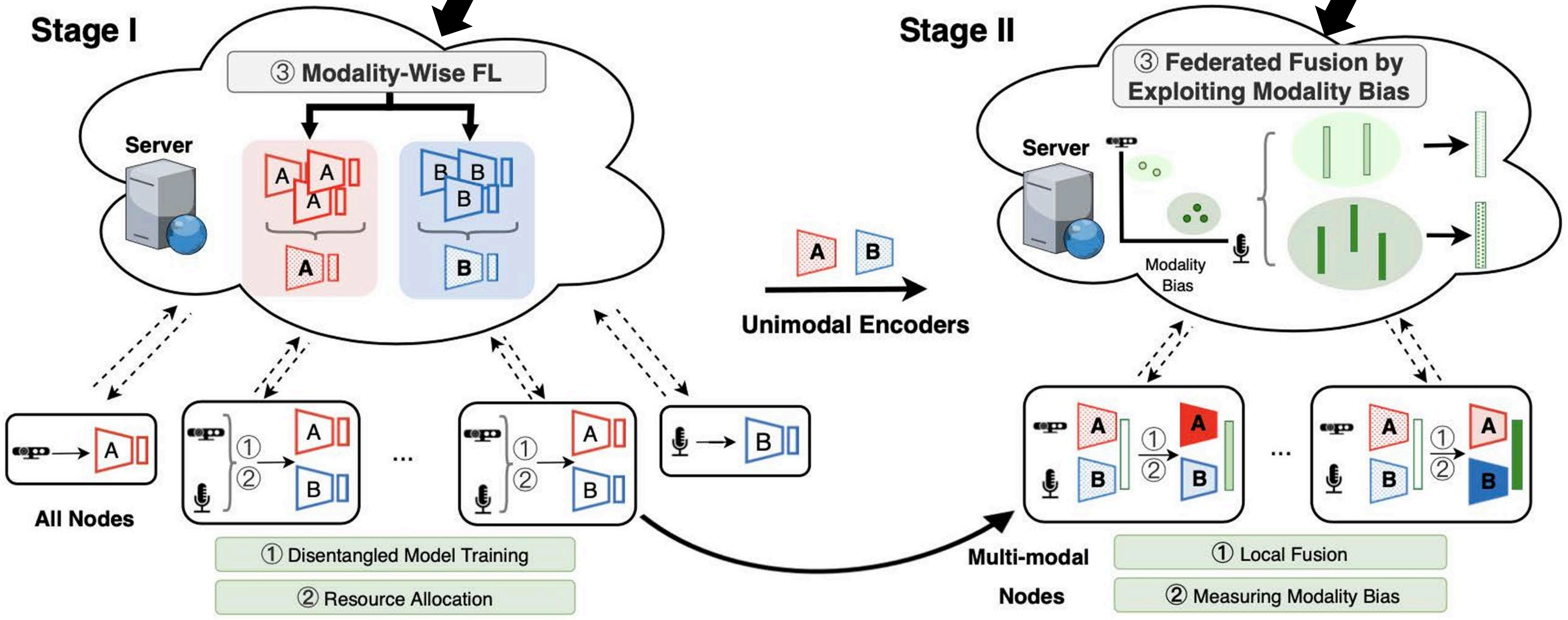
Figure 1: A typical application scenario of multi-modal federated learning systems: Alzheimer's Disease monitoring.

Ouyang et al. "Harmony: Heterogeneous Multi-Modal Federated Learning through Disentangled Model Training." Proceedings of the 21st Annual International Conference on Mobile Systems, Applications and (MobiSys), 2023

Harmony

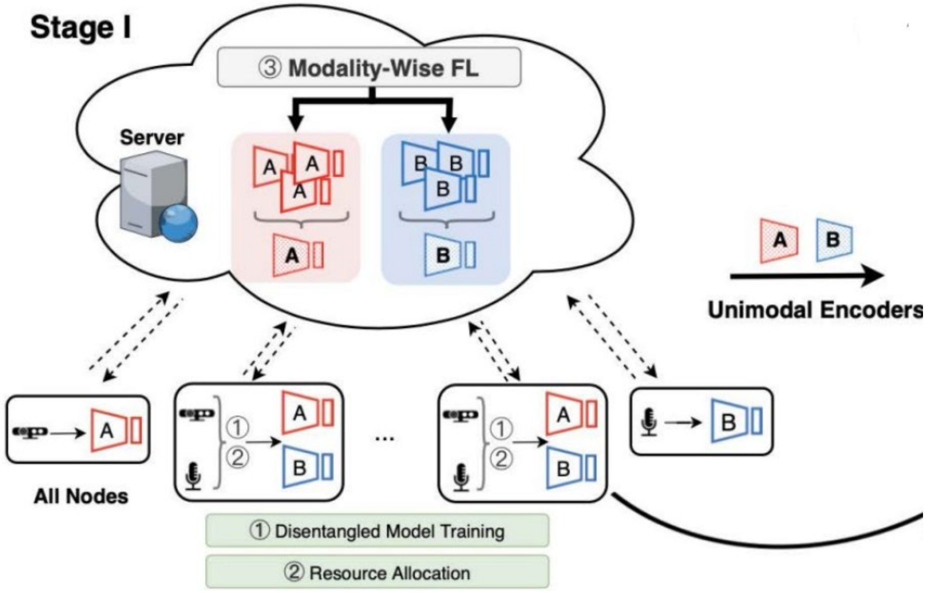
Multi-modal nodes train multiple single-modal networks.

The server clusters the nodes according to the modality biases and aggregates the classifier in each cluster.

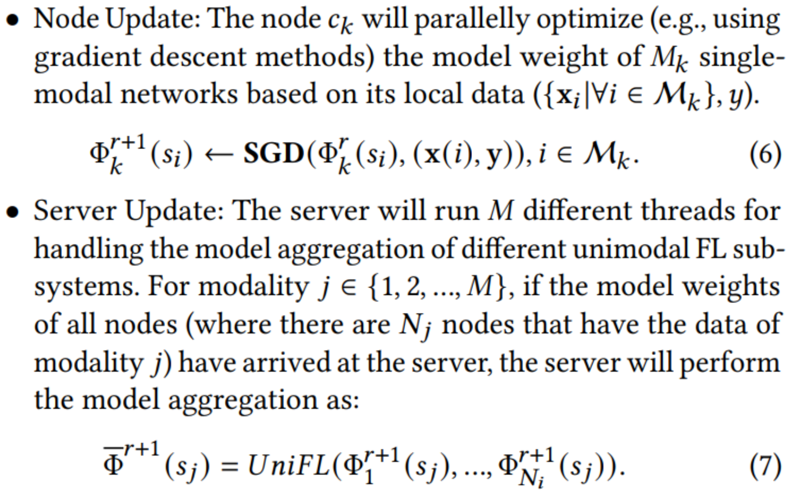


Ouyang et al. “Harmony: Heterogeneous Multi-Modal Federated Learning through Disentangled Model Training.” Proceedings of the 21st Annual International Conference on Mobile Systems, Applications and (MobiSys), 2023

Harmony



- Disentangled Model Training: The multi-modal nodes will train multiple single-modal networks rather than multi-modal fusion networks.
- Parallel Unimodal Federated Learning: After disentangling the training of multi-model models, all nodes will train and upload single-modal networks in modality-wise FL



- Node Update: The node c_k will parallelly optimize (e.g., using gradient descent methods) the model weight of M_k single-modal networks based on its local data $(\{\mathbf{x}_i | \forall i \in \mathcal{M}_k\}, y)$.

$$\Phi_k^{r+1}(s_i) \leftarrow \mathbf{SGD}(\Phi_k^r(s_i), (\mathbf{x}(i), \mathbf{y})), i \in \mathcal{M}_k. \qquad (6)$$

- Server Update: The server will run M different threads for handling the model aggregation of different unimodal FL subsystems. For modality $j \in \{1, 2, ..., M\}$, if the model weights of all nodes (where there are N_j nodes that have the data of modality j) have arrived at the server, the server will perform the model aggregation as:

$$\overline{\Phi}^{r+1}(s_j) = UniFL(\Phi_1^{r+1}(s_j), ..., \Phi_{N_i}^{r+1}(s_j)). \qquad (7)$$

PennState

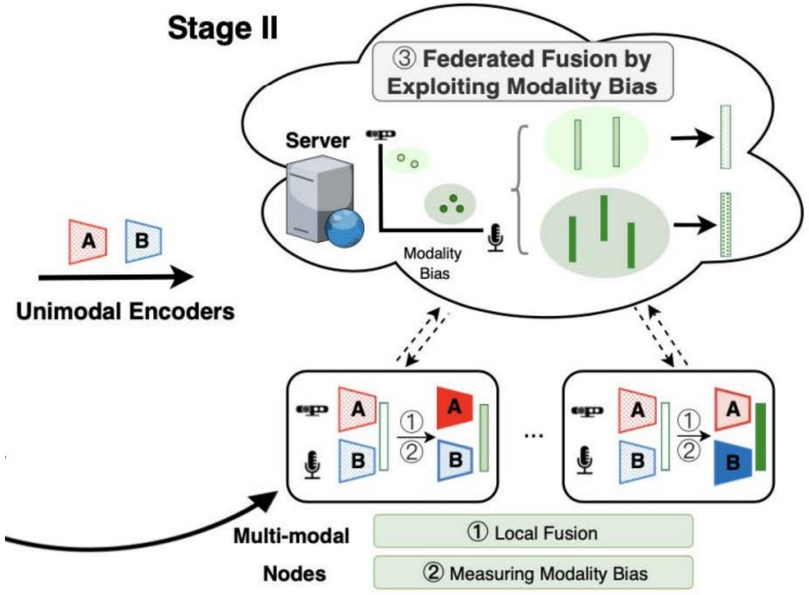
Ouyang et al. "Harmony: Heterogeneous Multi-Modal Federated Learning through Disentangled Model Training." Proceedings of the 21st Annual International Conference on Mobile Systems, Applications and (MobiSys), 2023

Harmony

- Measuring Modality Bias via Encoder Discrepancy: the multi-modal networks of different nodes may show substantial bias toward different modalities. They propose to measure and leverage such modality biases in different multi-modal networks.

$$d_k^r(i) = dis(f_{k,enc_i}^r(\cdot), f_{enc_i}^0(\cdot)). \qquad (9)$$

Here $dis(\cdot)$ measures the cosine distance of two weight vectors.



- Cluster-based Fusion Aggregation: the server will cluster the nodes according to their modality biases and aggregate the classifier layers with each cluster.
  - First normalize the encoder discrepancy value of each modality among all nodes.
  - K-means cluster: the server will aggregate the classifiers of multi-modal nodes within the same cluster.

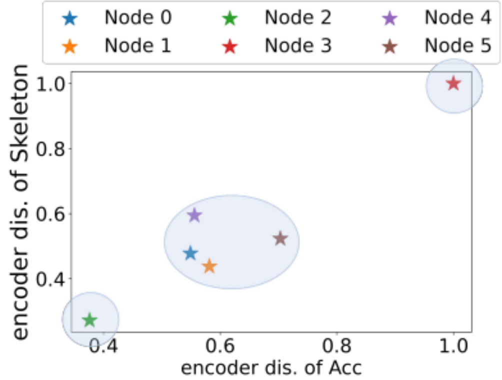
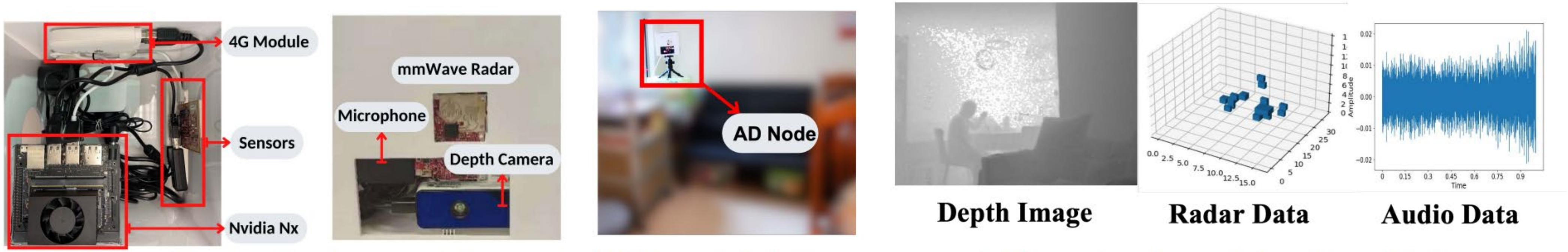


Figure 7: Visualization of encoder discrepancy vectors of multi-modal nodes. The nodes are grouped into three clusters based on the encoder discrepancy.

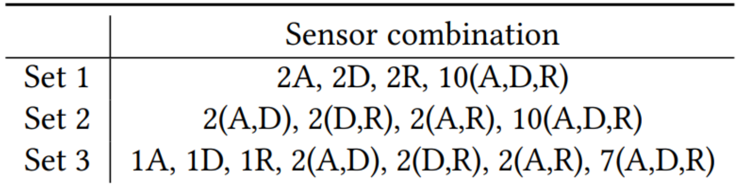
Ouyang et al. “Harmony: Heterogeneous Multi-Modal Federated Learning through Disentangled Model Training.” Proceedings of the 21st Annual International Conference on Mobile Systems, Applications and (MobiSys), 2023

Harmony



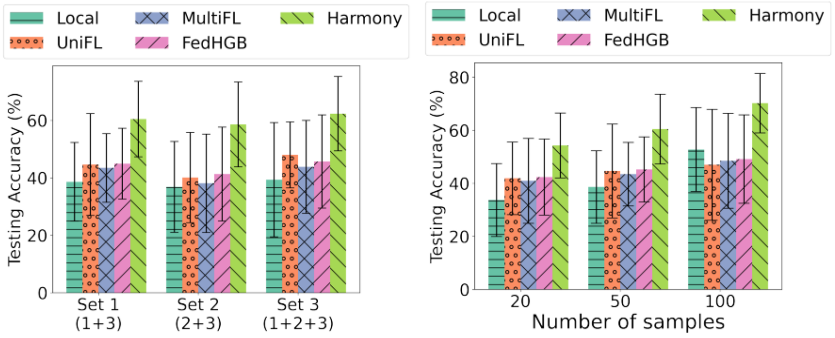
(a) Our multi-sensor hardware prototype. (b) Home installations. (c) Examples of recorded multi-modal data.

Figure 8: Our real-world multi-modal sensor testbed for Alzheimer's Disease monitoring. The nodes incorporating three sensor modalities (depth, mmWave radar, and audio) are deployed in the homes of 16 elderly subjects.



	Sensor combination
Set 1	2A, 2D, 2R, 10(A,D,R)
Set 2	2(A,D), 2(D,R), 2(A,R), 10(A,D,R)
Set 3	1A, 1D, 1R, 2(A,D), 2(D,R), 2(A,R), 7(A,D,R)

Table 1: Selected sensor combinations on 16 nodes. A, D, R denotes Audio, Depth, Radar, respectively, and 7(A,D,R) means seven nodes having three modalities.



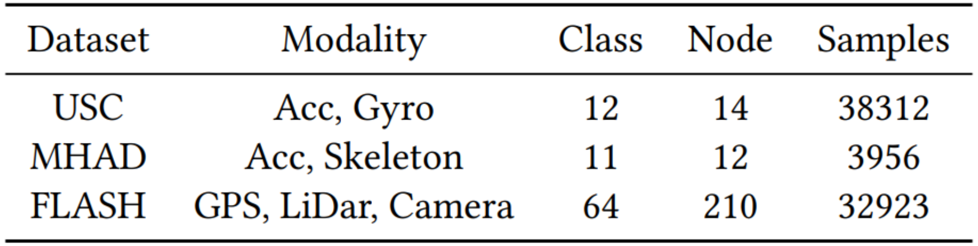
(a) Different modality sets. (b) Different amounts of data.

Figure 9: Accuracy performance on real-world multi-modal data. Harmony outperforms by 20% in mean accuracy over the baselines under various settings.

PennState

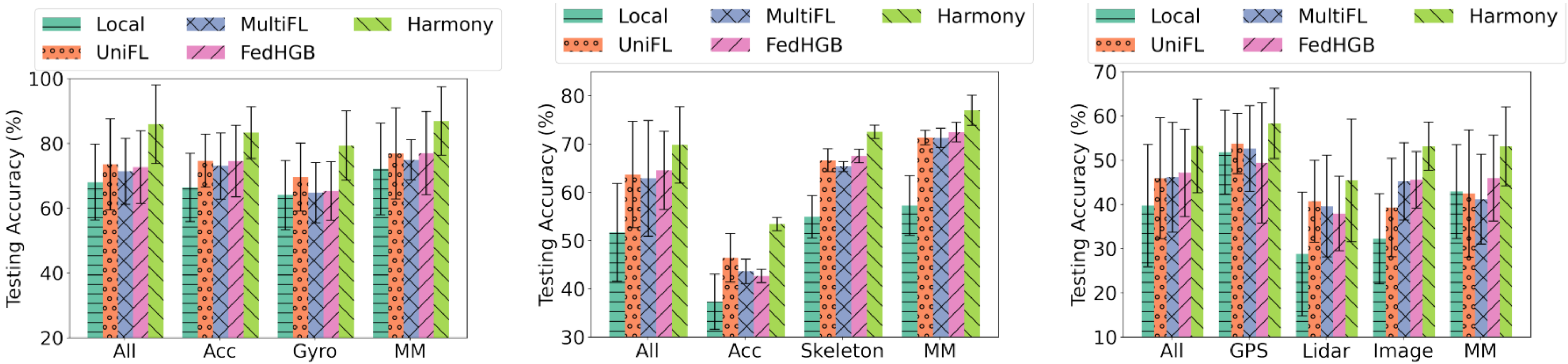
Ouyang et al. "Harmony: Heterogeneous Multi-Modal Federated Learning through Disentangled Model Training." Proceedings of the 21st Annual International Conference on Mobile Systems, Applications and (MobiSys), 2023

Harmony



Dataset	Modality	Class	Node	Samples
USC	Acc, Gyro	12	14	38312
MHAD	Acc, Skeleton	11	12	3956
FLASH	GPS, LiDar, Camera	64	210	32923

Table 2: Summary of the three multi-modal datasets.



(a) USC **(b) MHAD** **(c) FLASH**

Figure 12: Comparison of accuracy performance on different multi-modal datasets. Harmony consistently outperforms the state-of-the-art baselines for nodes with different data modalities.

PennState

Ouyang et al. "Harmony: Heterogeneous Multi-Modal Federated Learning through Disentangled Model Training." Proceedings of the 21st Annual International Conference on Mobile Systems, Applications and (MobiSys), 2023

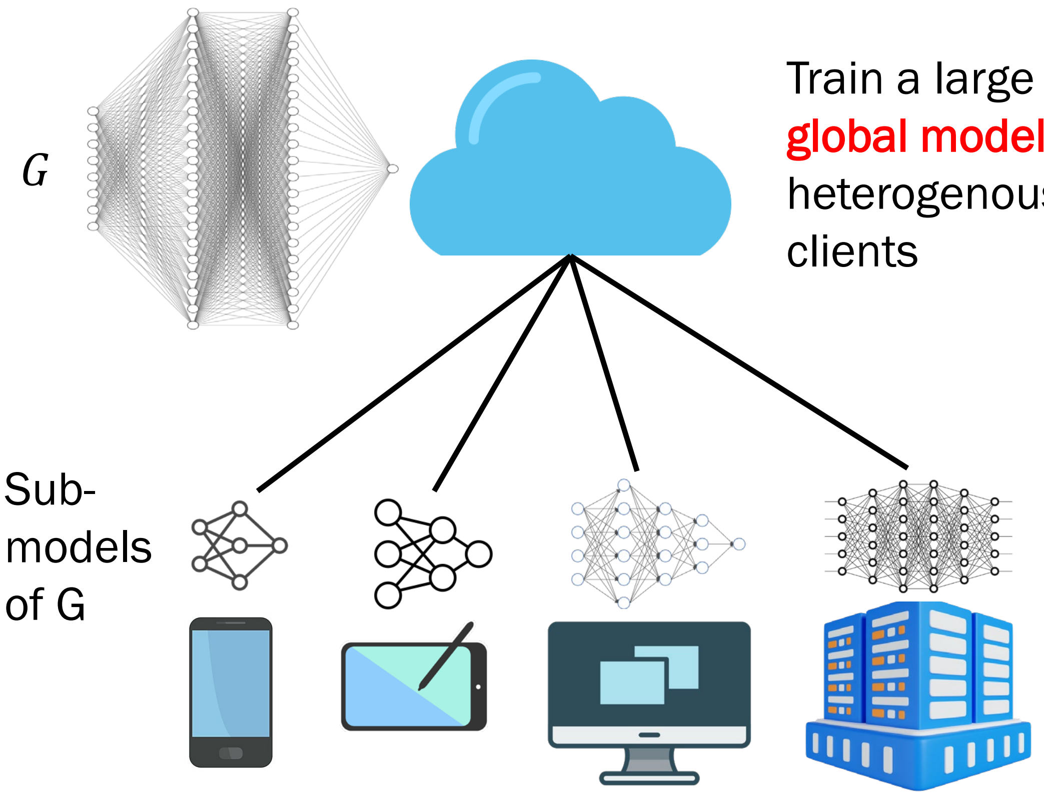
Approaches

- Regularization
  - FedProx
- Clustering
  - FedSEM
- Data Augmentation
  - FedCovid
- Multimodal Disentanglement
  - Harmony

Part 3

- Part 1: Federated Learning Introduction
- Part 2: Data/Statistical Heterogeneity
- Part 3: Model Heterogeneity
- Part 4: System Heterogeneity
- Part 5: Conclusion and Future Work

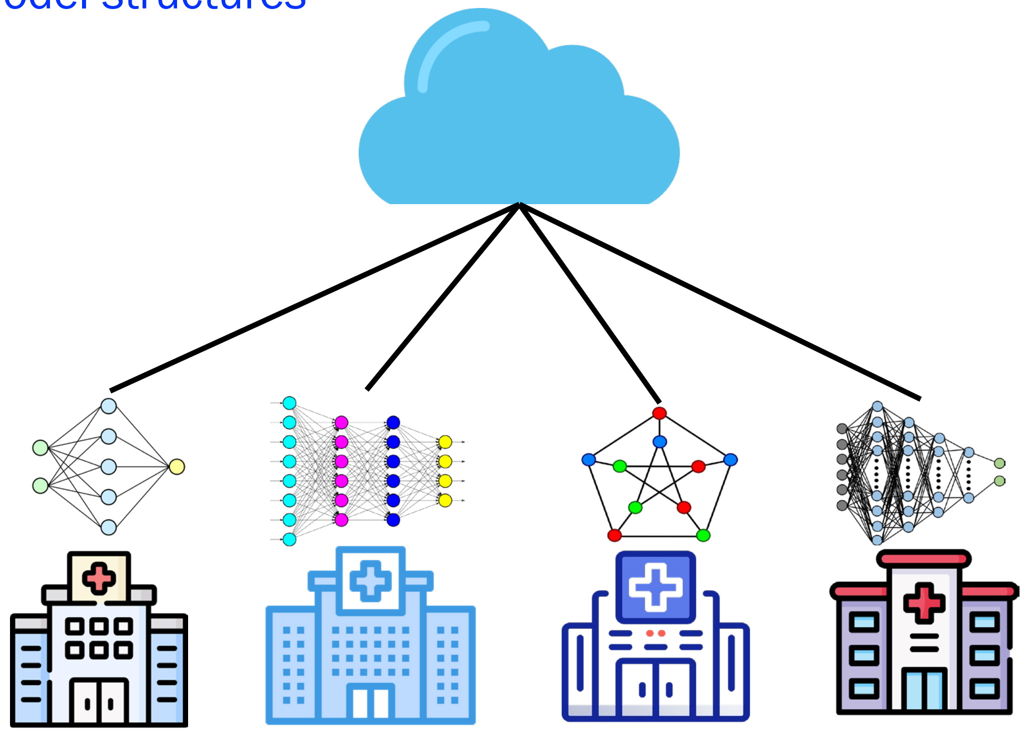
Model Heterogeneity



Train a large **global model** with heterogenous clients

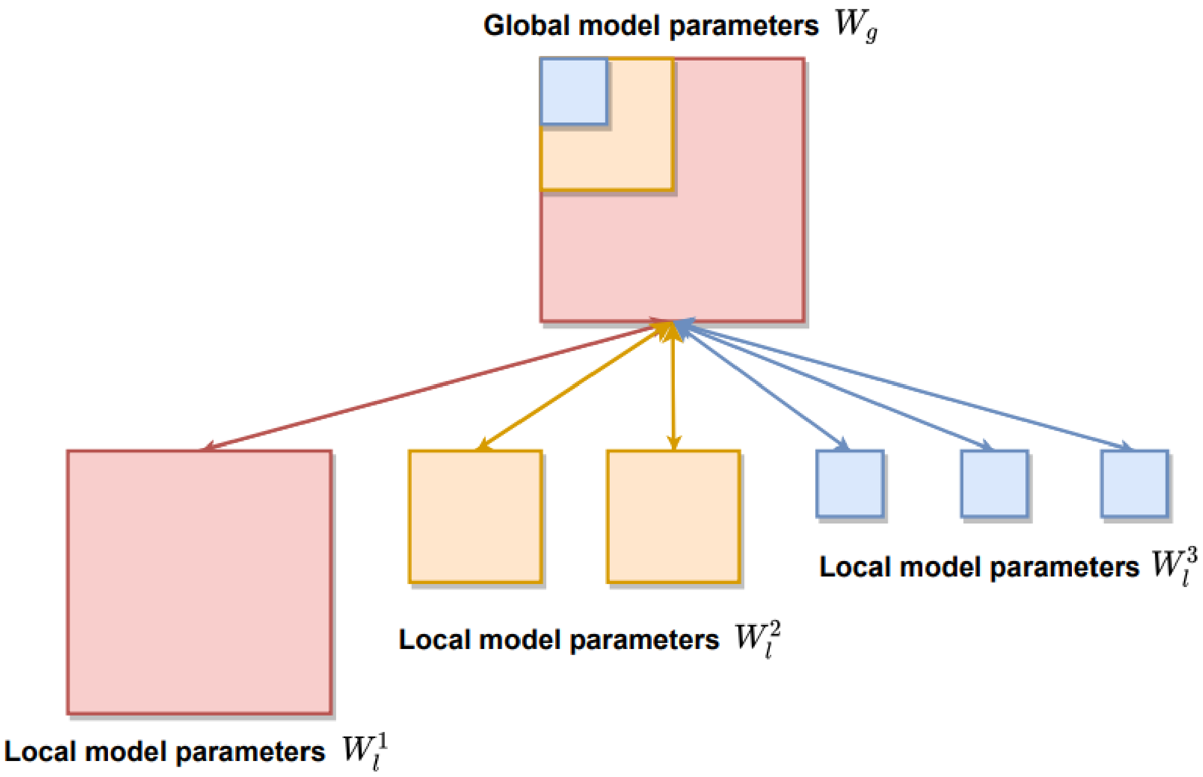
Sub-model training (partial heterogeneity)

Enhance the performance of each **client model** through collaborative learning without modifying client model structures



Heterogeneous model aggregation (complete heterogeneity)

HeteroFL


Global model parameters W_g
Local model parameters W_l^1
Local model parameters W_l^2
Local model parameters W_l^3

- Based on different clients' capacity, the server sends different sizes of the models to the clients.
- HeteroFL does aggregation for each part according to the client participation.

$$W_l^p = \frac{1}{m}\sum_{i=1}^{m} W_i^p, \quad W_l^{p-1} \setminus W_l^p = \frac{1}{m-m_p}\sum_{i=1}^{m-m_p} W_i^{p-1} \setminus W_i^p, \ldots$$

$$W_l^1 \setminus W_l^2 = \frac{1}{m-m_{2:p}}\sum_{i=1}^{m-m_{2:p}} W_i^1 \setminus W_i^2$$

$$W_g = W_l^1 = W_l^p \cup (W_l^{p-1} \setminus W_l^p) \cup \cdots \cup (W_l^1 \setminus W_l^2)$$

In this example, there are 6 clients including a large client, 2 medium clients, and 3 small clients.

Diao et al. “HeteroFL: Computation and communication efficient federated learning for heterogeneous clients.” International Conference on Learning Representations, 2021.

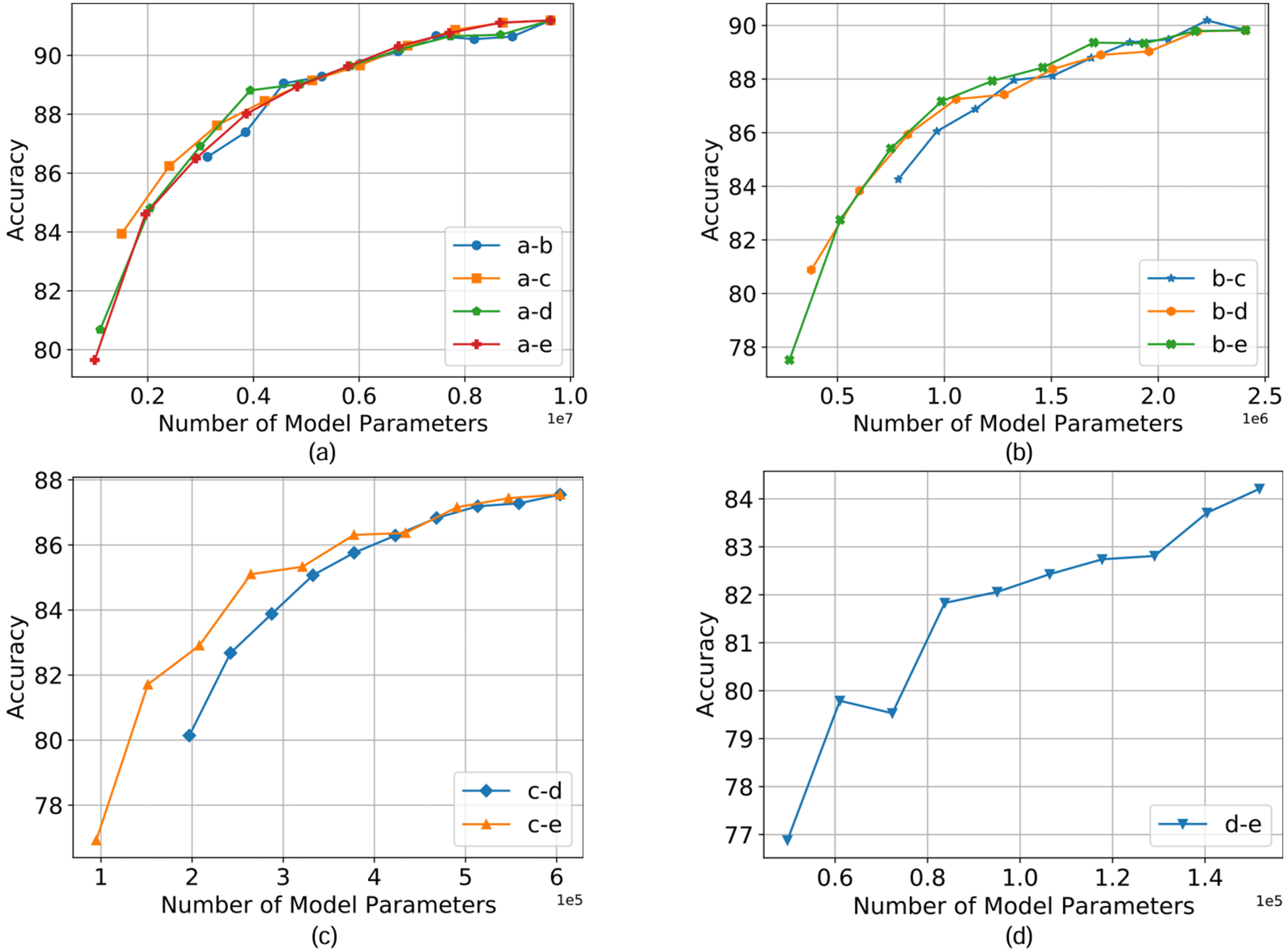
HeteroFL

- Computation complexity levels

a	1.0
b	0.5
c	0.25
d	0.125
e	0.0625

All the model parameters

≈Logistic regression



(a)

(b)

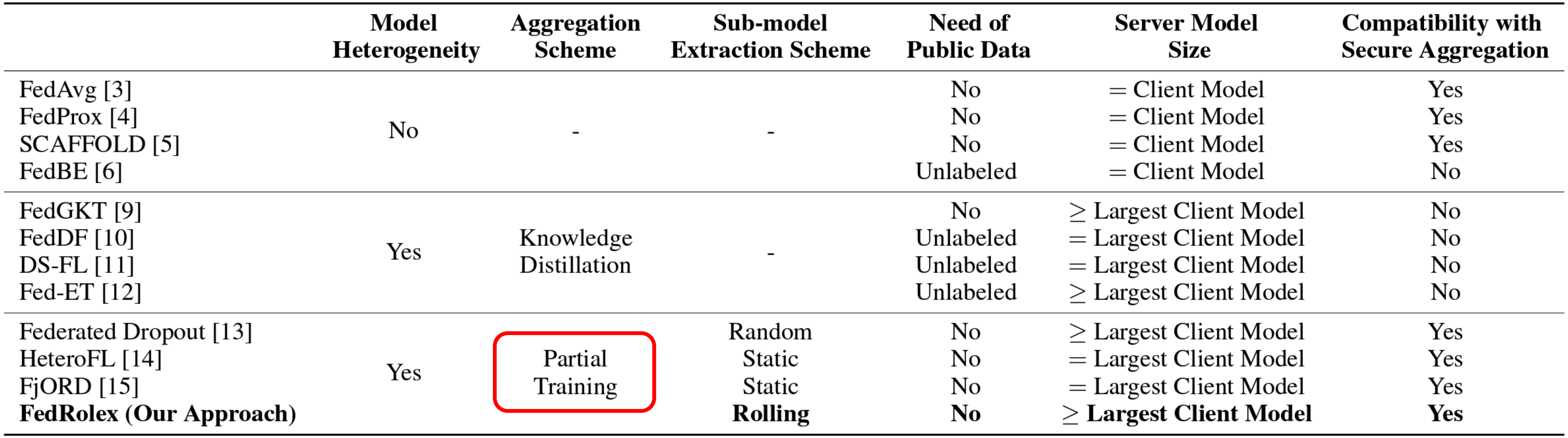
(c)

(d)

Figure 2: Interpolation experimental results for CIFAR10 (IID) dataset between global model complexity ((a) a, (b) b, (c) c, (d) d) and various smaller model complexities.

Diao et al. “HeteroFL: Computation and communication efficient federated learning for heterogeneous clients.” International Conference on Learning Representations, 2021.

FedRolex



	Model Heterogeneity	Aggregation Scheme	Sub-model Extraction Scheme	Need of Public Data	Server Model Size	Compatibility with Secure Aggregation
FedAvg [3]	No	-	-	No	$=$ Client Model	Yes
FedProx [4]				No	$=$ Client Model	Yes
SCAFFOLD [5]				No	$=$ Client Model	Yes
FedBE [6]				Unlabeled	$=$ Client Model	No
FedGKT [9]	Yes	Knowledge Distillation	-	No	$\geq$ Largest Client Model	No
FedDF [10]				Unlabeled	$=$ Largest Client Model	No
DS-FL [11]				Unlabeled	$=$ Largest Client Model	No
Fed-ET [12]				Unlabeled	$\geq$ Largest Client Model	No
Federated Dropout [13]	Yes	Partial Training	Random	No	$\geq$ Largest Client Model	Yes
HeteroFL [14]			Static	No	$=$ Largest Client Model	Yes
FjORD [15]			Static	No	$=$ Largest Client Model	Yes
FedRolex (Our Approach)			**Rolling**	**No**	$\geq$ **Largest Client Model**	**Yes**

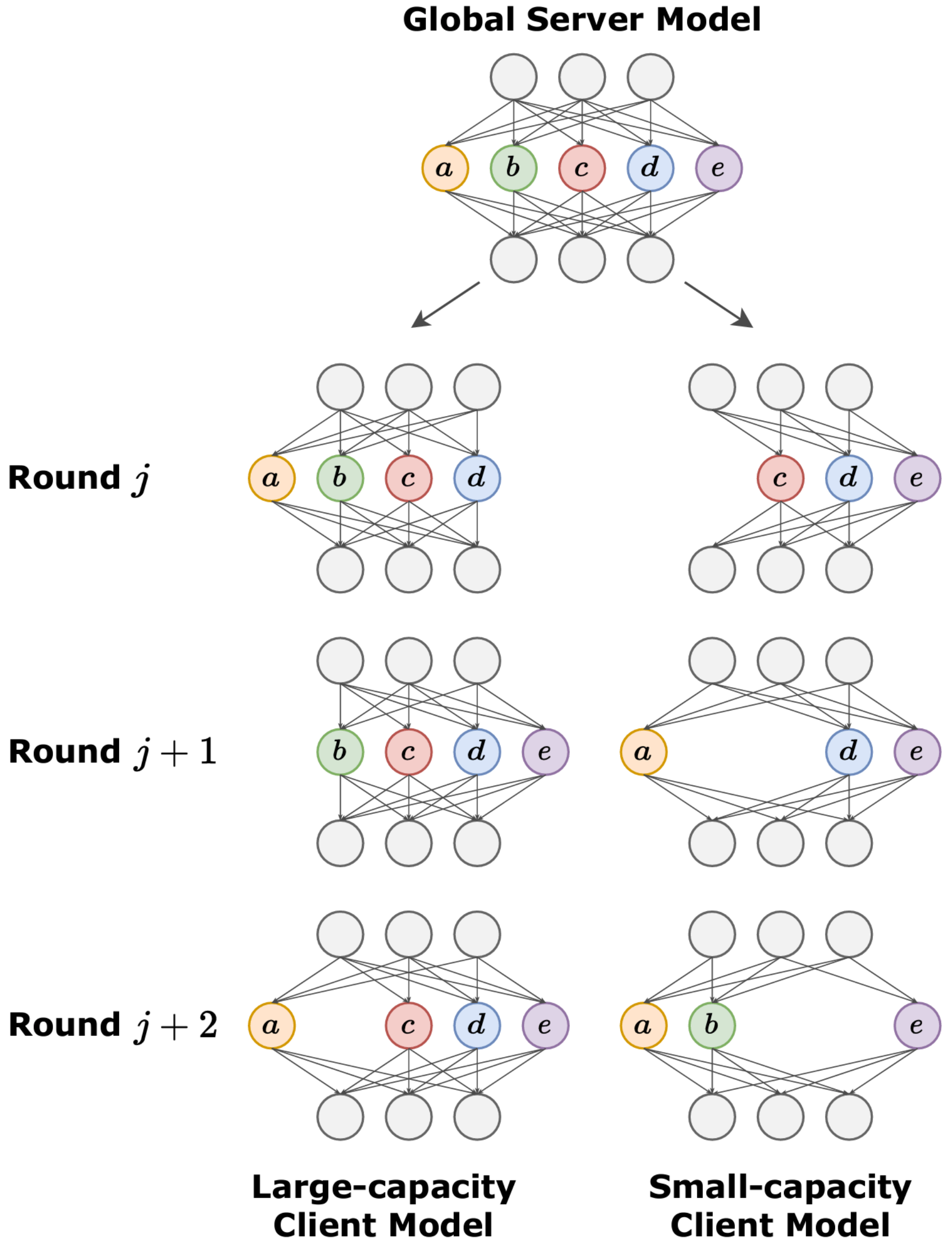
Existing PT-based methods: The sub-models are extracted in ways (either random or static) such that the parameters of the global server model are not evenly trained. This makes the server model vulnerable to client drift induced by the inconsistency between individual client model and server model architectures–a unique challenge of model-heterogeneous FL.

PennState

Alam et al. “FedRolex: Model-Heterogeneous Federated Learning with Rolling Sub-Model Extraction.” 36th Conference on Neural Information Processing Systems, 2022.

FedRolex

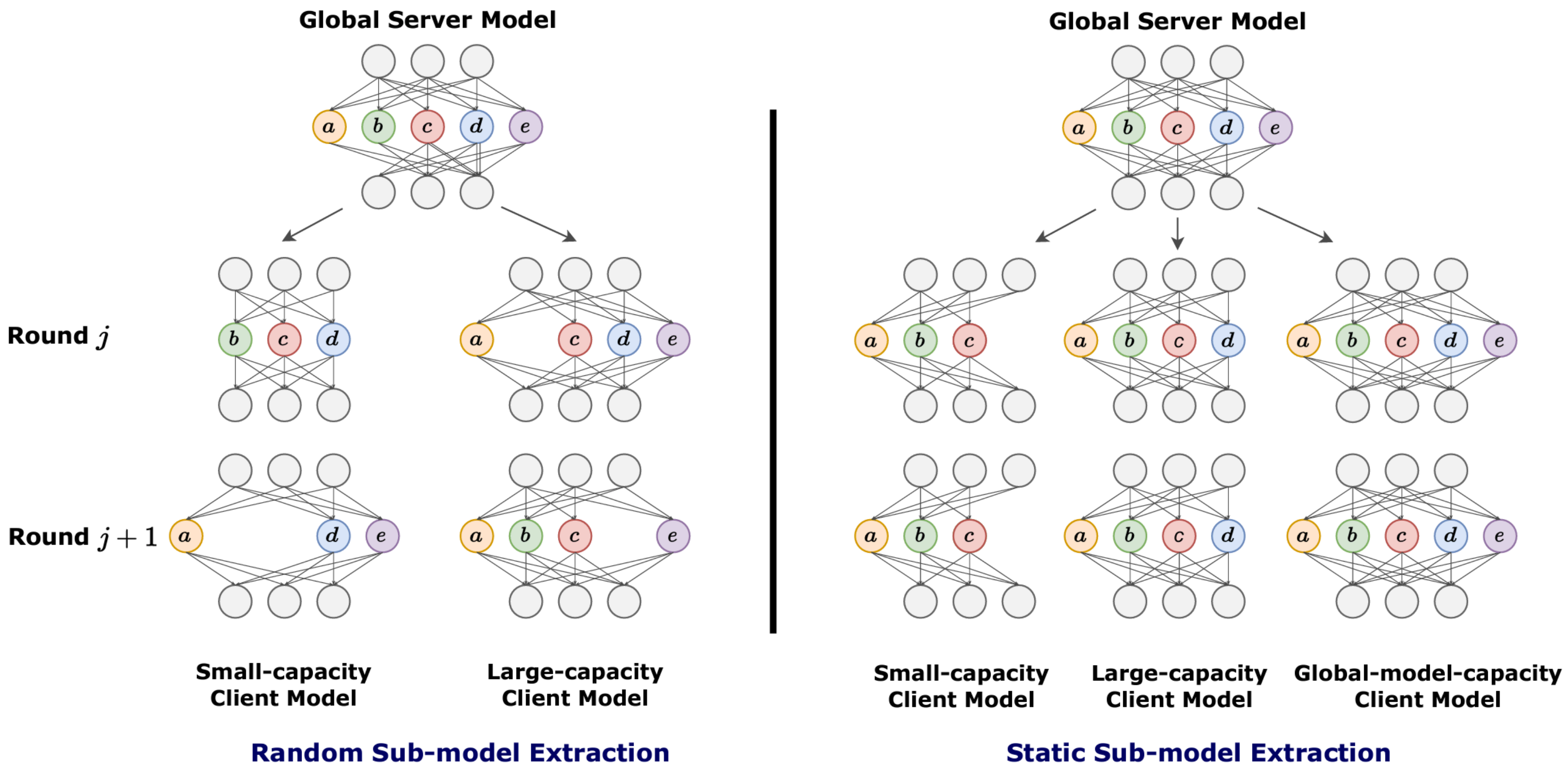
- Model-heterogeneous with rolling sub-model extraction.
- The aggregation still follows the FedAvg-based approach, which covers the overlapping and non-overlapping part.



Alam et al. "FedRolex: Model-Heterogeneous Federated Learning with Rolling Sub-Model Extraction." 36th Conference on Neural Information Processing Systems, 2022.

FedRolex

- Two sub-model extraction strategies


Global Server Model
a
b
c
d
e
Round j
Round $j+1$
Small-capacity Client Model
Large-capacity Client Model
Random Sub-model Extraction
Global Server Model
Small-capacity Client Model
Large-capacity Client Model
Global-model-capacity Client Model
Static Sub-model Extraction

Alam et al. “FedRolex: Model-Heterogeneous Federated Learning with Rolling Sub-Model Extraction.” 36th Conference on Neural Information Processing Systems, 2022.

FedRolex

- Global model accuracy

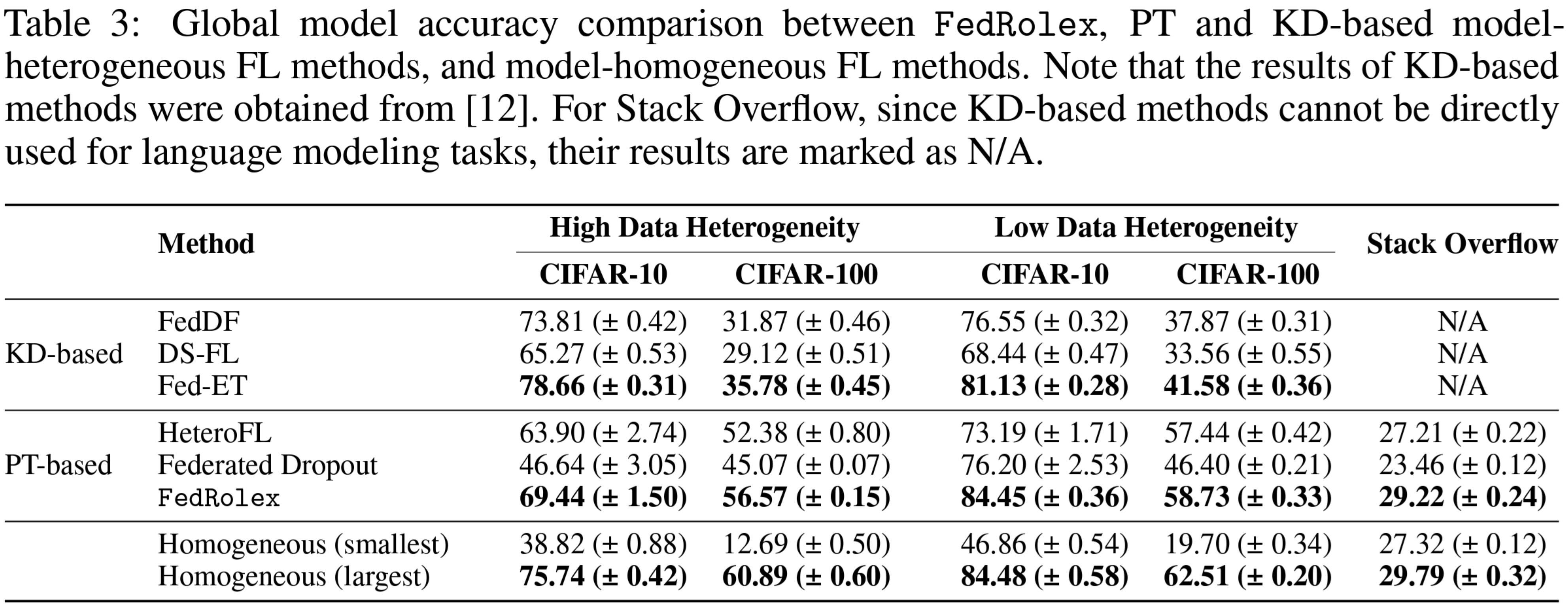


Table 3: Global model accuracy comparison between `FedRolex`, PT and KD-based model-heterogeneous FL methods, and model-homogeneous FL methods. Note that the results of KD-based methods were obtained from [12]. For Stack Overflow, since KD-based methods cannot be directly used for language modeling tasks, their results are marked as N/A.

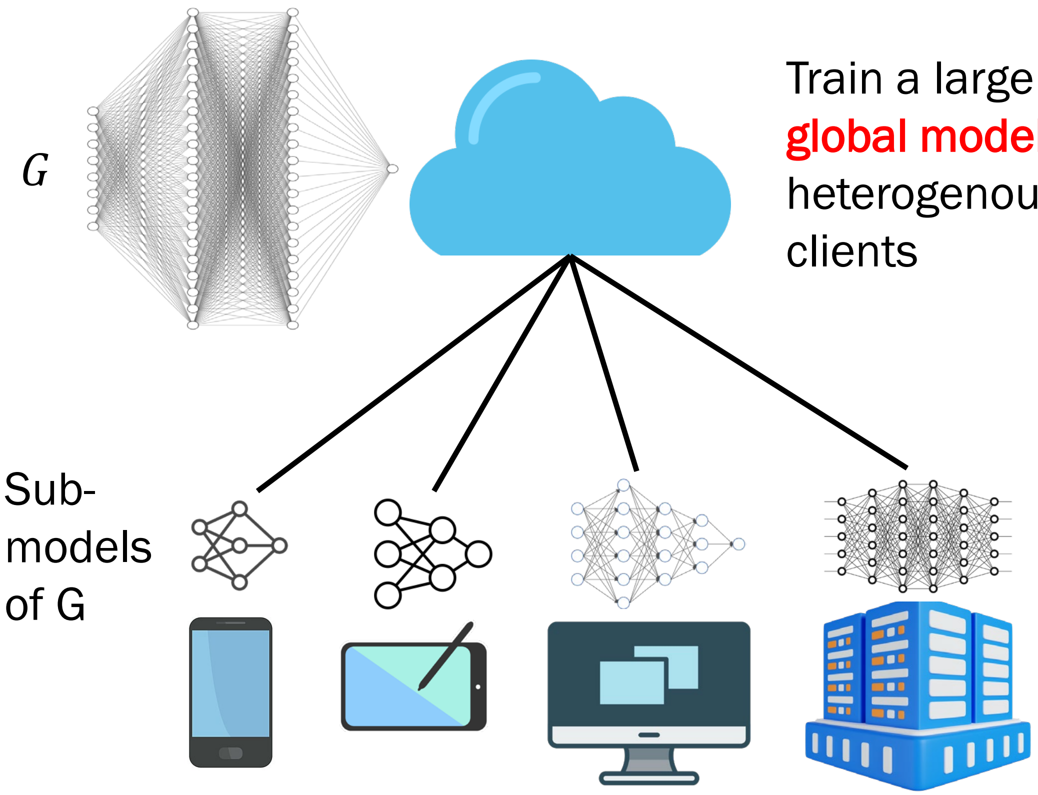
	Method	**High Data Heterogeneity**		**Low Data Heterogeneity**		**Stack Overflow**
		CIFAR-10	**CIFAR-100**	**CIFAR-10**	**CIFAR-100**	
KD-based	FedDF	73.81 (± 0.42)	31.87 (± 0.46)	76.55 (± 0.32)	37.87 (± 0.31)	N/A
	DS-FL	65.27 (± 0.53)	29.12 (± 0.51)	68.44 (± 0.47)	33.56 (± 0.55)	N/A
	Fed-ET	**78.66 (± 0.31)**	**35.78 (± 0.45)**	**81.13 (± 0.28)**	**41.58 (± 0.36)**	N/A
PT-based	HeteroFL	63.90 (± 2.74)	52.38 (± 0.80)	73.19 (± 1.71)	57.44 (± 0.42)	27.21 (± 0.22)
	Federated Dropout	46.64 (± 3.05)	45.07 (± 0.07)	76.20 (± 2.53)	46.40 (± 0.21)	23.46 (± 0.12)
	`FedRolex`	**69.44 (± 1.50)**	**56.57 (± 0.15)**	**84.45 (± 0.36)**	**58.73 (± 0.33)**	**29.22 (± 0.24)**
	Homogeneous (smallest)	38.82 (± 0.88)	12.69 (± 0.50)	46.86 (± 0.54)	19.70 (± 0.34)	27.32 (± 0.12)
	Homogeneous (largest)	**75.74 (± 0.42)**	**60.89 (± 0.60)**	**84.48 (± 0.58)**	**62.51 (± 0.20)**	**29.79 (± 0.32)**

Alam et al. “FedRolex: Model-Heterogeneous Federated Learning with Rolling Sub-Model Extraction.” 36th Conference on Neural Information Processing Systems, 2022.

Summary of Partial Heterogeneity

- Strong constraints of the clients' models' structures. Clients may not be able to utilize their models freely. The core ideas are:
  - Contribute to one global model by partial training at different clients.
  - Share the identical part, which is used as the carrier of the information exchange.

Model Heterogeneity



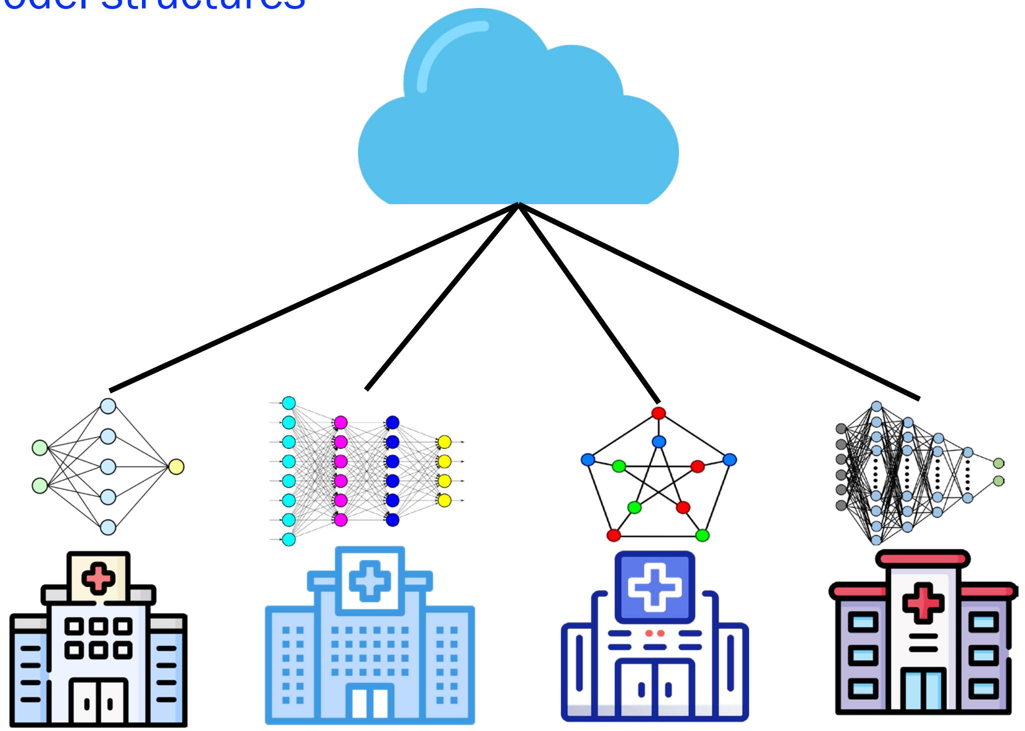
G

Train a large global model with heterogenous clients

Sub-models of G

Sub-model training (partial heterogeneity)

Enhance the performance of each client model through collaborative learning without modifying client model structures



Heterogeneous model aggregation (complete heterogeneity)

FedGH

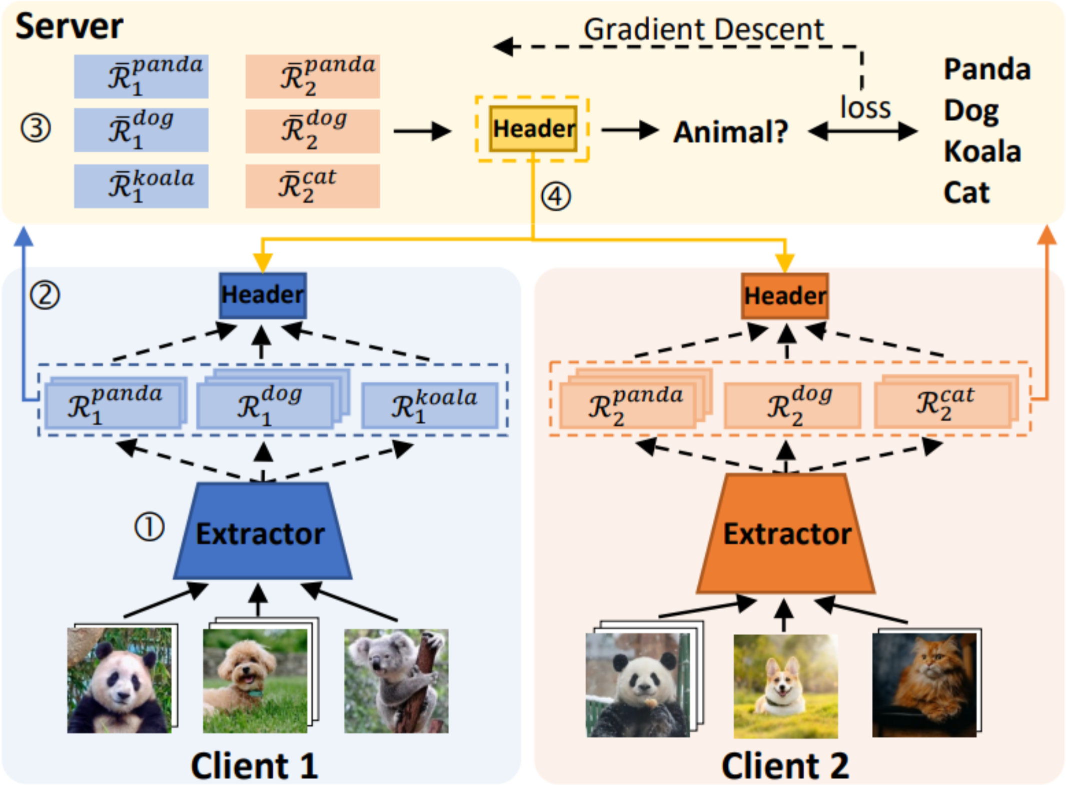


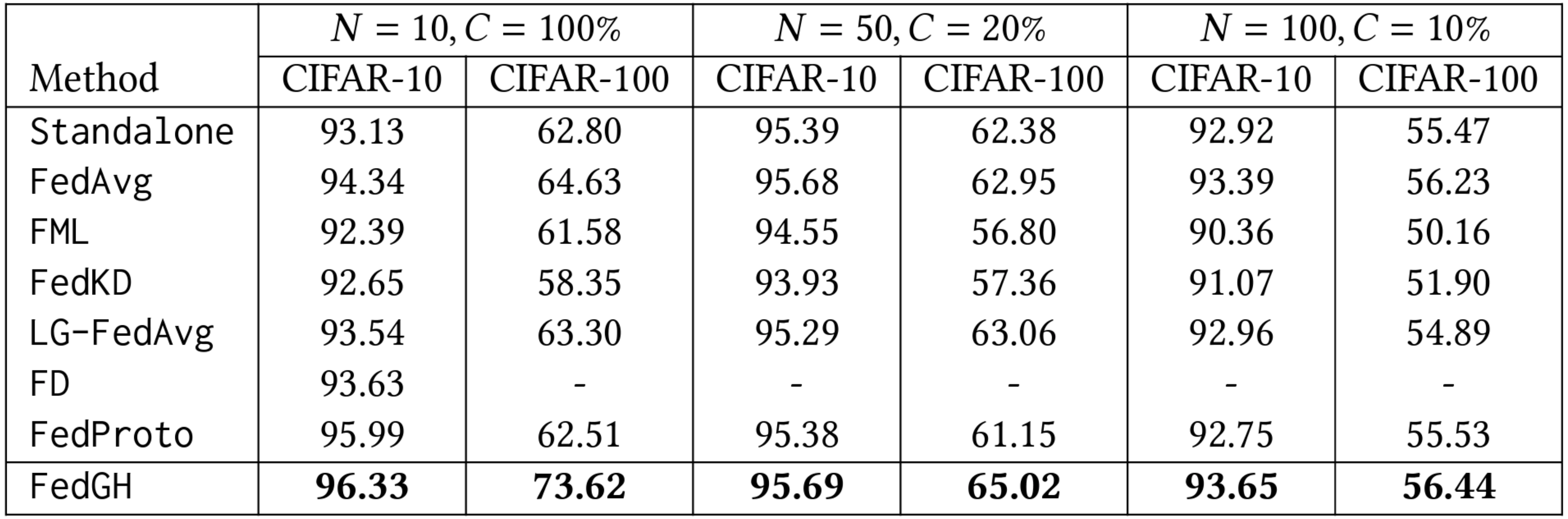
Figure 1: The workflow of the proposed FedGH approach.

- Clients share the identical header and have their own feature extractors.
- The header will be transmitted between the server and the clients.
- The information of the classes and their representation need to be uploaded to update the global header.

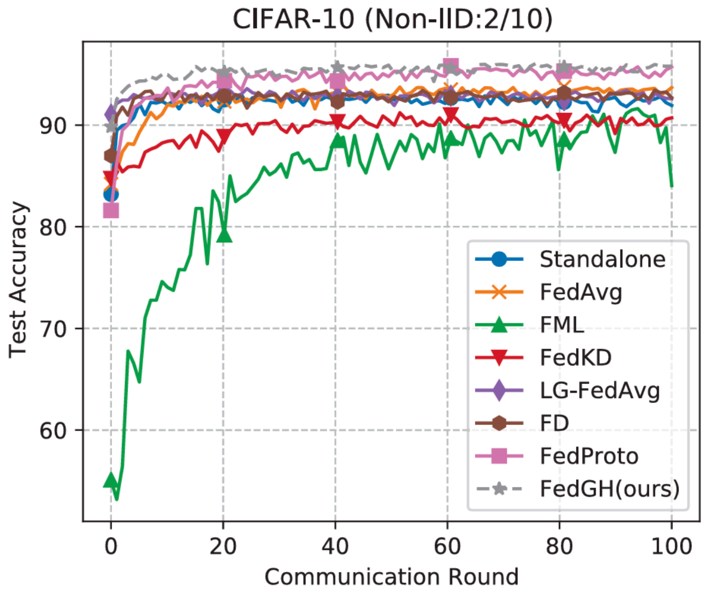
Yi et al. "FedGH: Heterogeneous Federated Learning with Generalized Global Header." Proceedings of the 31st ACM International Conference on Multimedia, 2023.

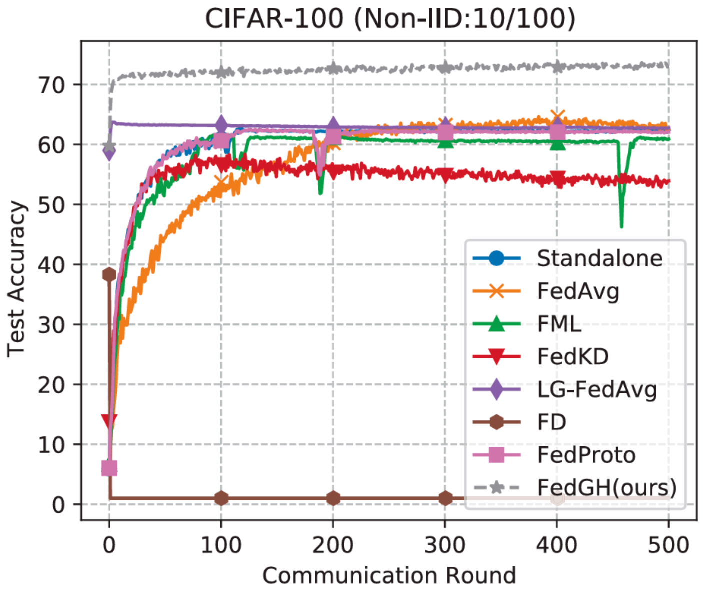
FedGH

- Results



Method	$N=10, C=100\%$		$N=50, C=20\%$		$N=100, C=10\%$	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
Standalone	93.13	62.80	95.39	62.38	92.92	55.47
FedAvg	94.34	64.63	95.68	62.95	93.39	56.23
FML	92.39	61.58	94.55	56.80	90.36	50.16
FedKD	92.65	58.35	93.93	57.36	91.07	51.90
LG-FedAvg	93.54	63.30	95.29	63.06	92.96	54.89
FD	93.63	-	-	-	-	-
FedProto	95.99	62.51	95.38	61.15	92.75	55.53
FedGH	**96.33**	**73.62**	**95.69**	**65.02**	**93.65**	**56.44**





Yi et al. "FedGH: Heterogeneous Federated Learning with Generalized Global Header." Proceedings of the 31st ACM International Conference on Multimedia, 2023.

FedGH

- The header only contains limited information, leading to unsatisfactory performance.
- Uploading representations and class labels may have privacy concerns.

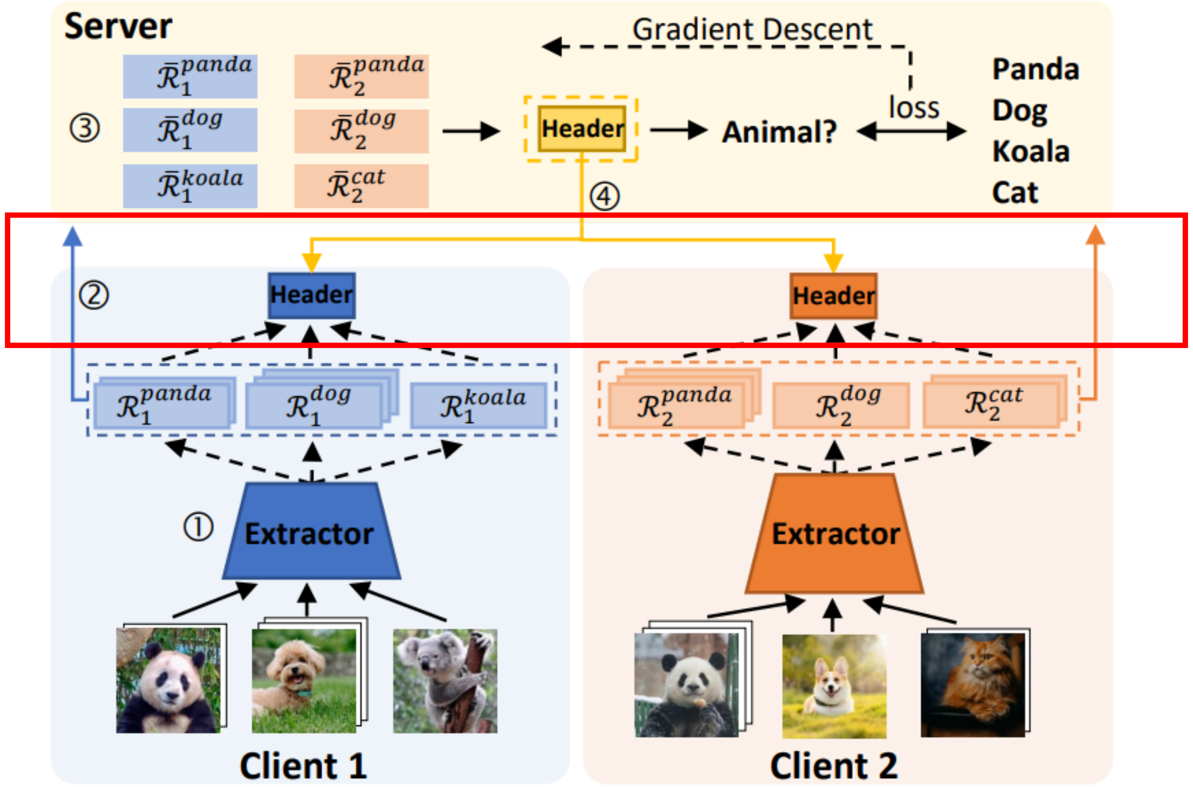
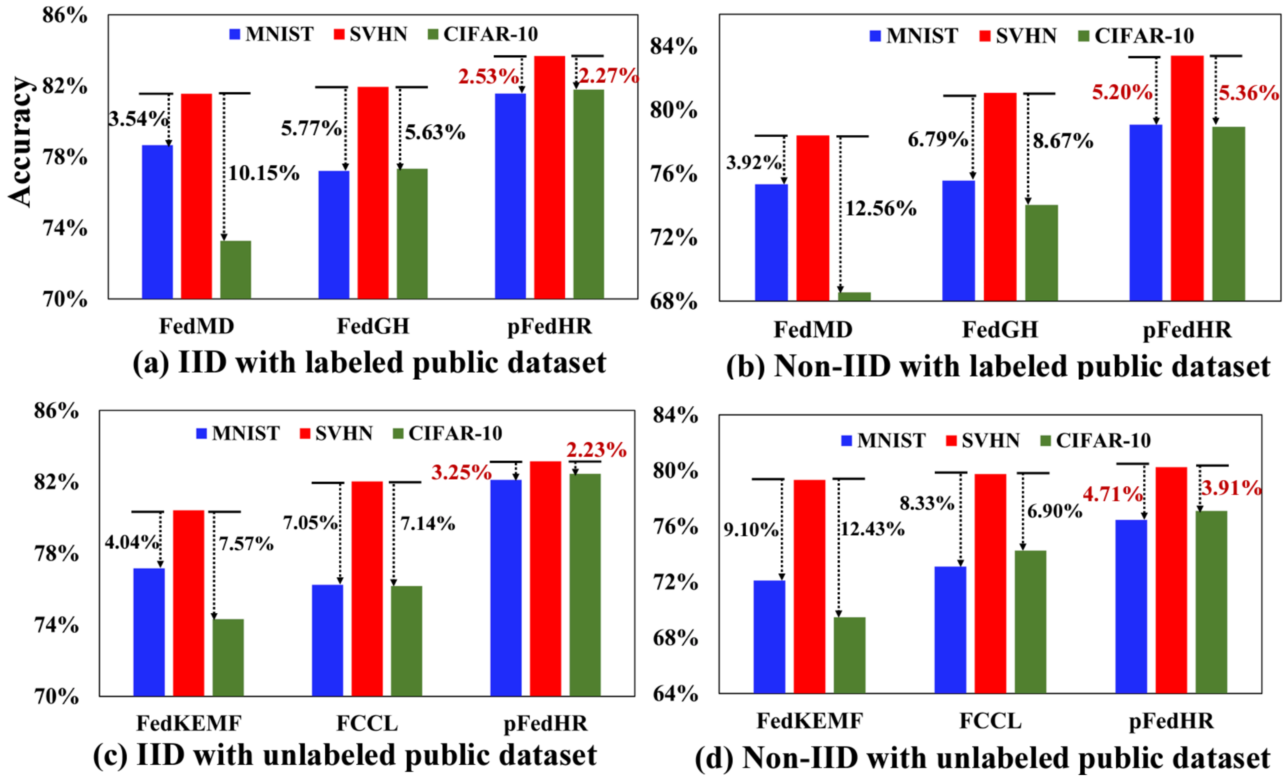


Figure 1: The workflow of the proposed FedGH approach.

Yi et al. "FedGH: Heterogeneous Federated Learning with Generalized Global Header." Proceedings of the 31st ACM International Conference on Multimedia, 2023.

pFedHR

- Public data usage



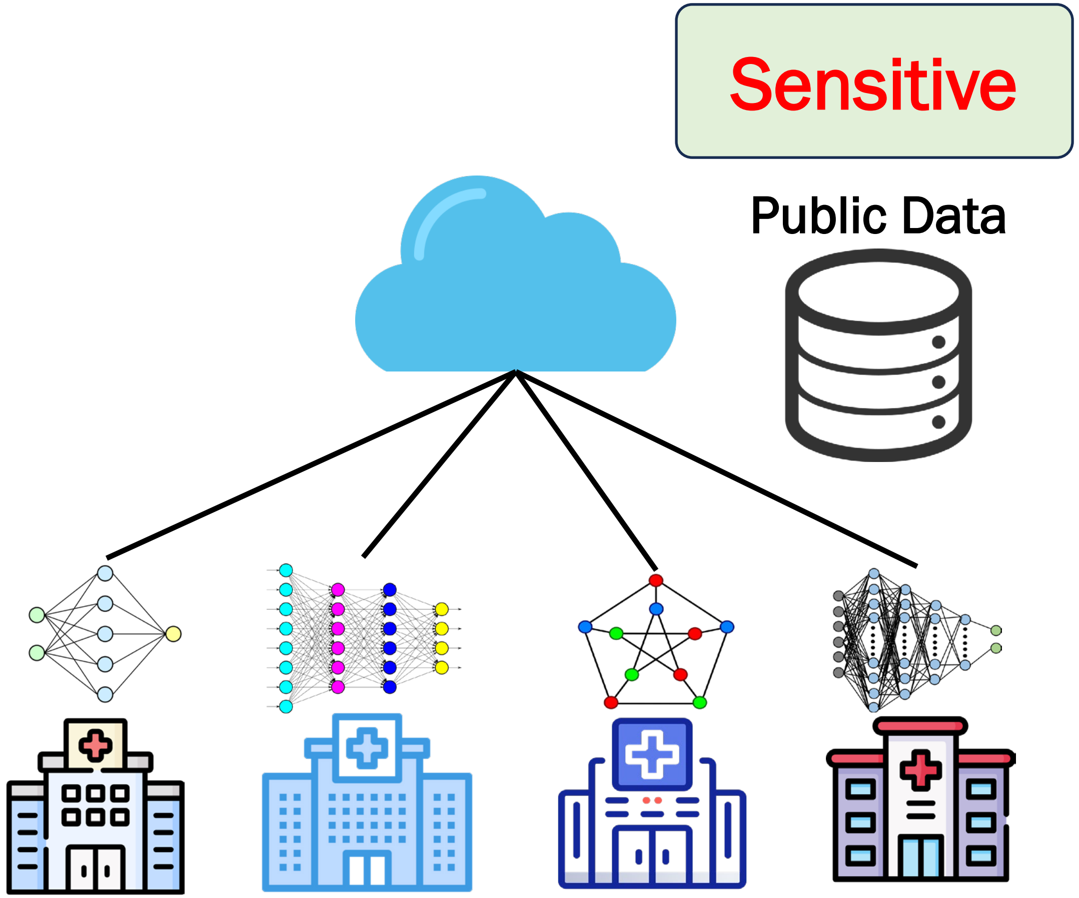
(a) IID with labeled public dataset

(b) Non-IID with labeled public dataset

(c) IID with unlabeled public dataset

(d) Non-IID with unlabeled public dataset

Training on the SVHN dataset with different public data.

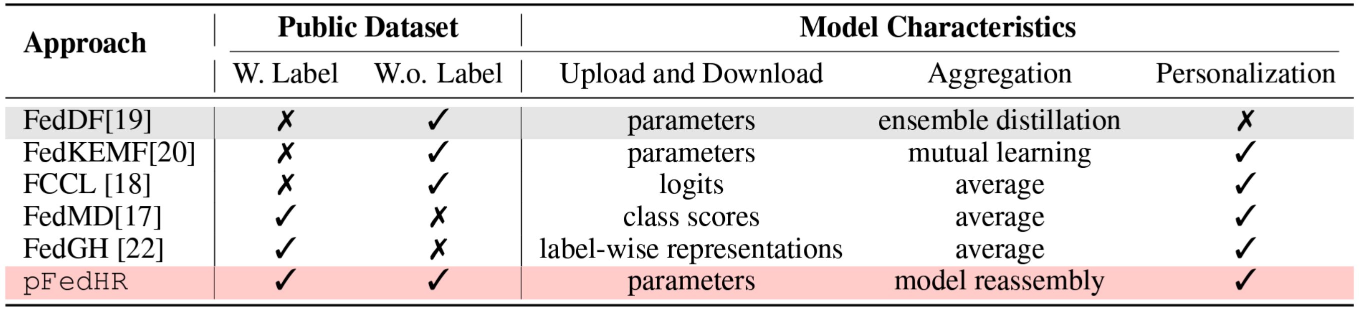


Sensitive

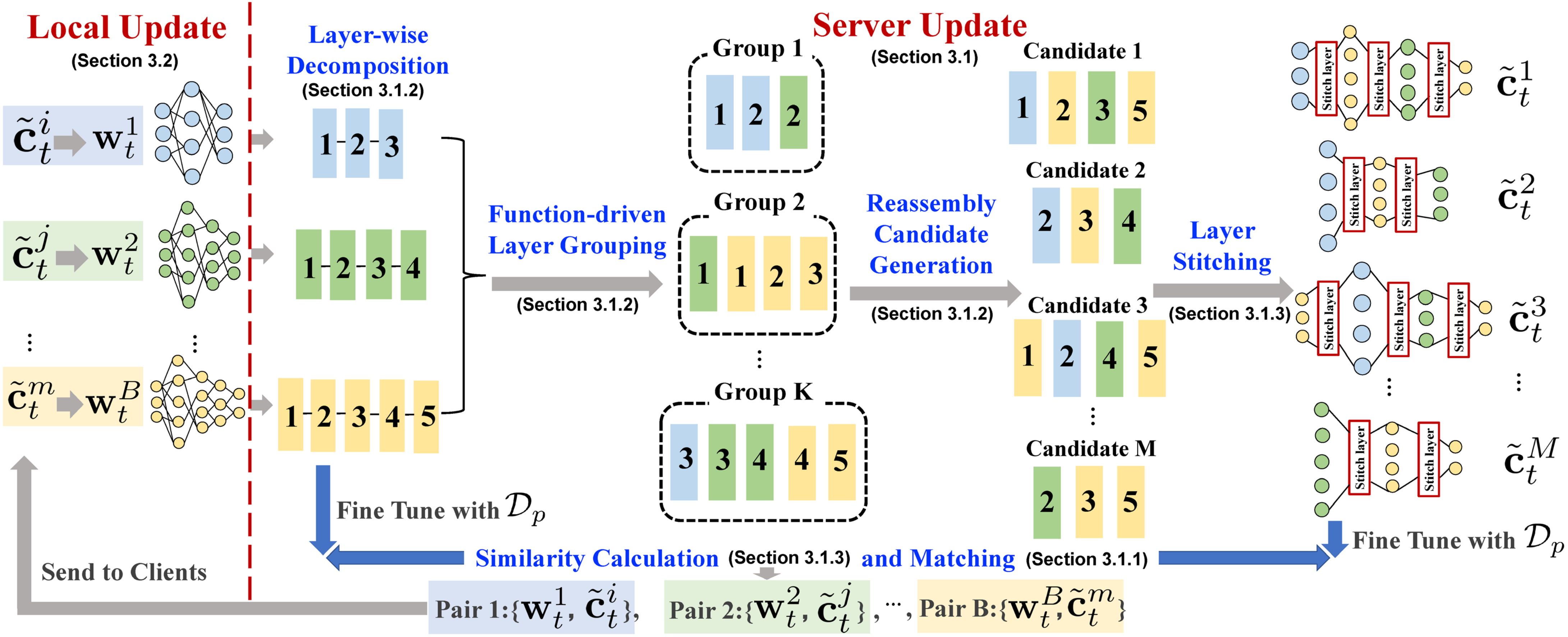
Public Data

Wang et al. “Towards personalized federated learning via heterogeneous model reassembly.” 37th Conference on Neural Information Processing Systems, 2023.

pFedHR

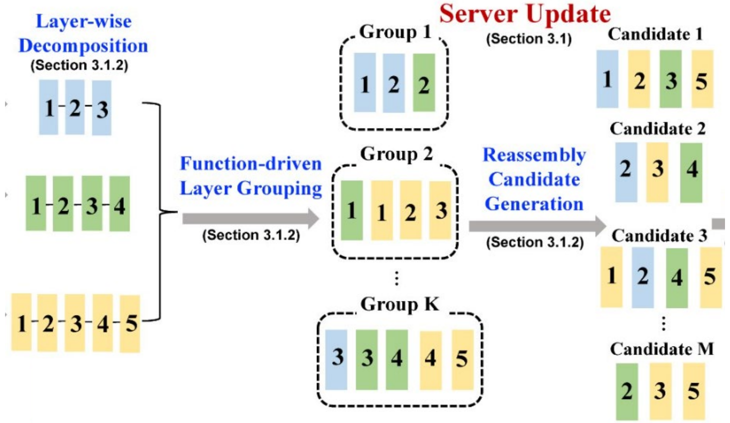


Approach	Public Dataset		Model Characteristics		
	W. Label	W.o. Label	Upload and Download	Aggregation	Personalization
FedDF[19]	✗	✓	parameters	ensemble distillation	✗
FedKEMF[20]	✗	✓	parameters	mutual learning	✓
FCCL [18]	✗	✓	logits	average	✓
FedMD[17]	✓	✗	class scores	average	✓
FedGH [22]	✓	✗	label-wise representations	average	✓
pFedHR	✓	✓	parameters	model reassembly	✓



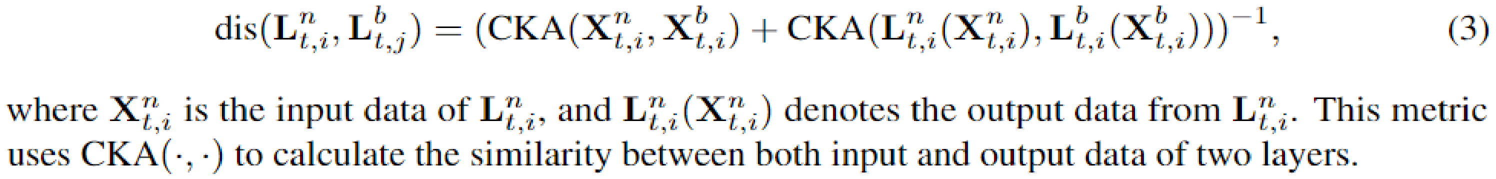
Local Update
(Section 3.2)
$\tilde{\mathbf{c}}_t^i \rightarrow \mathbf{w}_t^1$
$\tilde{\mathbf{c}}_t^j \rightarrow \mathbf{w}_t^2$
$\tilde{\mathbf{c}}_t^m \rightarrow \mathbf{w}_t^B$
Layer-wise Decomposition
(Section 3.1.2)
1-2-3
1-2-3-4
1-2-3-4-5
Function-driven Layer Grouping
(Section 3.1.2)
Group 1
1 2 2
Group 2
1 1 2 3
Group K
3 3 4 4 5
Server Update
(Section 3.1)
Reassembly Candidate Generation
(Section 3.1.2)
Candidate 1
1 2 3 5
Candidate 2
2 3 4
Candidate 3
1 2 4 5
Candidate M
2 3 5
Layer Stitching
(Section 3.1.3)
Stitch layer
$\tilde{\mathbf{c}}_t^1$
$\tilde{\mathbf{c}}_t^2$
$\tilde{\mathbf{c}}_t^3$
$\tilde{\mathbf{c}}_t^M$
Fine Tune with $\mathcal{D}_p$
Fine Tune with $\mathcal{D}_p$
Similarity Calculation (Section 3.1.3) and Matching (Section 3.1.1)
Send to Clients
Pair 1:$\{\mathbf{w}_t^1, \tilde{\mathbf{c}}_t^i\}$, Pair 2:$\{\mathbf{w}_t^2, \tilde{\mathbf{c}}_t^j\}$, ···, Pair B:$\{\mathbf{w}_t^B, \tilde{\mathbf{c}}_t^m\}$

Wang et al. “Towards personalized federated learning via heterogeneous model reassembly.” 37th Conference on Neural Information Processing Systems, 2023.



pFedHR

- Layer-wise Decomposition
- Function-driven Layer Grouping
  - Measure the distance between each layers via CKA (centered kernel alignment)



$$\mathrm{dis}(\mathbf{L}^n_{t,i}, \mathbf{L}^b_{t,j}) = (\mathrm{CKA}(\mathbf{X}^n_{t,i}, \mathbf{X}^b_{t,i}) + \mathrm{CKA}(\mathbf{L}^n_{t,i}(\mathbf{X}^n_{t,i}), \mathbf{L}^b_{t,i}(\mathbf{X}^b_{t,i})))^{-1}, \qquad (3)$$

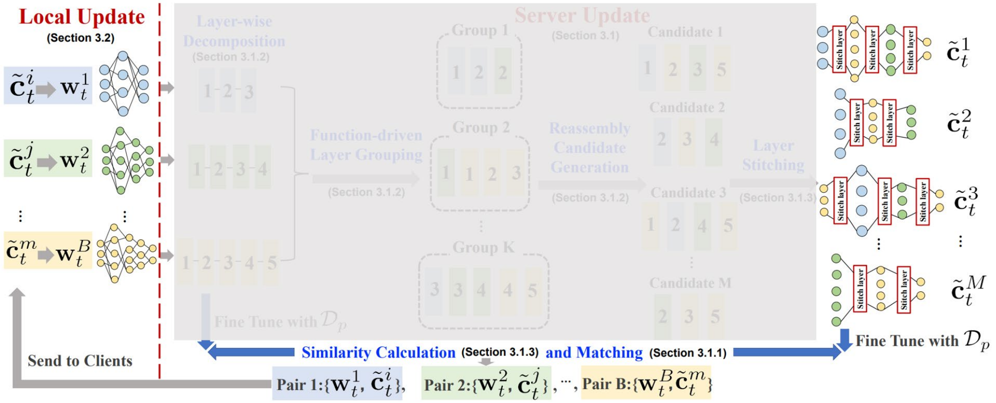
where $\mathbf{X}^n_{t,i}$ is the input data of $\mathbf{L}^n_{t,i}$, and $\mathbf{L}^n_{t,i}(\mathbf{X}^n_{t,i})$ denotes the output data from $\mathbf{L}^n_{t,i}$. This metric uses $\mathrm{CKA}(\cdot, \cdot)$ to calculate the similarity between both input and output data of two layers.

  - Conduct K-means-style algorithm to group layers of B models into K clusters.

- Reassembly Candidate Generation
  - All the operation types should be included
  - All the defined functions should be included
  - The layer order should follow the natural order

Wang et al. “Towards personalized federated learning via heterogeneous model reassembly.” 37th Conference on Neural Information Processing Systems, 2023.

pFedHR


Local Update
(Section 3.2)
$\tilde{\mathbf{c}}_t^i \rightarrow \mathbf{w}_t^1$
$\tilde{\mathbf{c}}_t^j \rightarrow \mathbf{w}_t^2$
$\tilde{\mathbf{c}}_t^m \rightarrow \mathbf{w}_t^B$
Send to Clients
Server Update
(Section 3.1)
Layer-wise Decomposition
(Section 3.1.2)
1-2-3
1-2-3-4
1-2-3-4-5
Fine Tune with $\mathcal{D}_p$
Function-driven Layer Grouping
(Section 3.1.2)
Group 1
1 2 2
Group 2
1 1 2 3
Group K
3 3 4 4 5
Reassembly Candidate Generation
(Section 3.1.2)
Candidate 1
1 2 3 5
Candidate 2
2 3 4
Candidate 3
1 2 4 5
Candidate M
2 3 5
Layer Stitching
(Section 3.1.3)
Stitch layer
$\tilde{\mathbf{c}}_t^1$
$\tilde{\mathbf{c}}_t^2$
$\tilde{\mathbf{c}}_t^3$
$\tilde{\mathbf{c}}_t^M$
Fine Tune with $\mathcal{D}_p$
Similarity Calculation (Section 3.1.3) and Matching (Section 3.1.1)
Pair 1:$\{\mathbf{w}_t^1, \tilde{\mathbf{c}}_t^i\}$, Pair 2:$\{\mathbf{w}_t^2, \tilde{\mathbf{c}}_t^j\}$, ⋯, Pair B:$\{\mathbf{w}_t^B, \tilde{\mathbf{c}}_t^m\}$

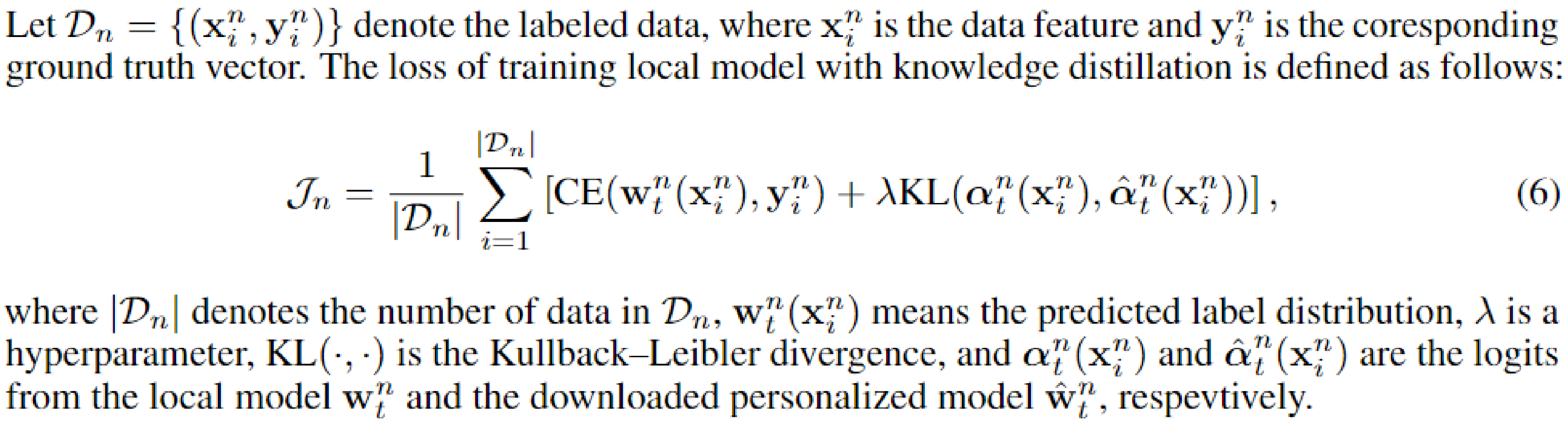
- Layer stitching
  - We apply a simple MLP as the stitching layer to match the different dimensions of two consecutive layers.
  - The simple MLP can also control the number of the parameters and maintain more information from the original models as much as possible.
- Similarity calculation
  - We need to select the best fitting teacher to guide the local model learning at the next communication round. In this case, we calculate the similarity of the logits from each pair of the local models and the candidate models:

$$\mathrm{sim}(\mathbf{w}_t^n, \mathbf{c}_t^m; \mathcal{D}_p) = \mathrm{sim}(\mathbf{w}_t^n, \tilde{\mathbf{c}}_t^m; \mathcal{D}_p) = \frac{1}{P}\sum_{p=1}^{P} \cos(\boldsymbol{\alpha}_t^n(\mathbf{x}_p), \boldsymbol{\alpha}_t^m(\mathbf{x}_p)),$$

Wang et al. “Towards personalized federated learning via heterogeneous model reassembly.” 37th Conference on Neural Information Processing Systems, 2023.

pFedHR

- Client Update:



Let $\mathcal{D}_n = \{(\mathbf{x}_i^n, \mathbf{y}_i^n)\}$ denote the labeled data, where $\mathbf{x}_i^n$ is the data feature and $\mathbf{y}_i^n$ is the coresponding ground truth vector. The loss of training local model with knowledge distillation is defined as follows:

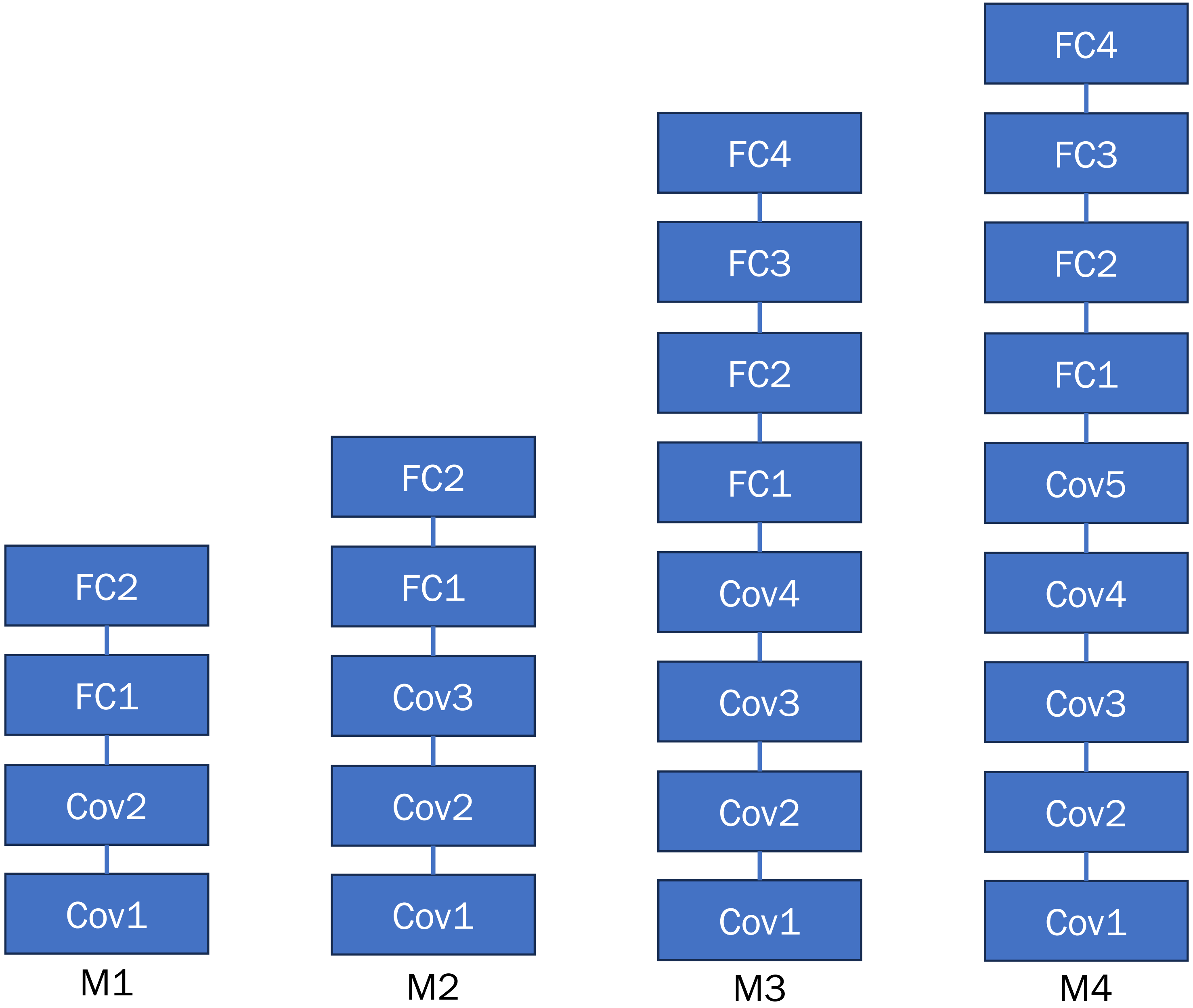
$$\mathcal{J}_n = \frac{1}{|\mathcal{D}_n|} \sum_{i=1}^{|\mathcal{D}_n|} \left[\mathrm{CE}(\mathbf{w}_t^n(\mathbf{x}_i^n), \mathbf{y}_i^n) + \lambda \mathrm{KL}(\boldsymbol{\alpha}_t^n(\mathbf{x}_i^n), \hat{\boldsymbol{\alpha}}_t^n(\mathbf{x}_i^n)) \right], \tag{6}$$

where $|\mathcal{D}_n|$ denotes the number of data in $\mathcal{D}_n$, $\mathbf{w}_t^n(\mathbf{x}_i^n)$ means the predicted label distribution, λ is a hyperparameter, $\mathrm{KL}(\cdot, \cdot)$ is the Kullback–Leibler divergence, and $\boldsymbol{\alpha}_t^n(\mathbf{x}_i^n)$ and $\hat{\boldsymbol{\alpha}}_t^n(\mathbf{x}_i^n)$ are the logits from the local model $\mathbf{w}_t^n$ and the downloaded personalized model $\hat{\mathbf{w}}_t^n$, respevtively.

Wang et al. “Towards personalized federated learning via heterogeneous model reassembly.” 37th Conference on Neural Information Processing Systems, 2023.

pFedHR

- Experiments

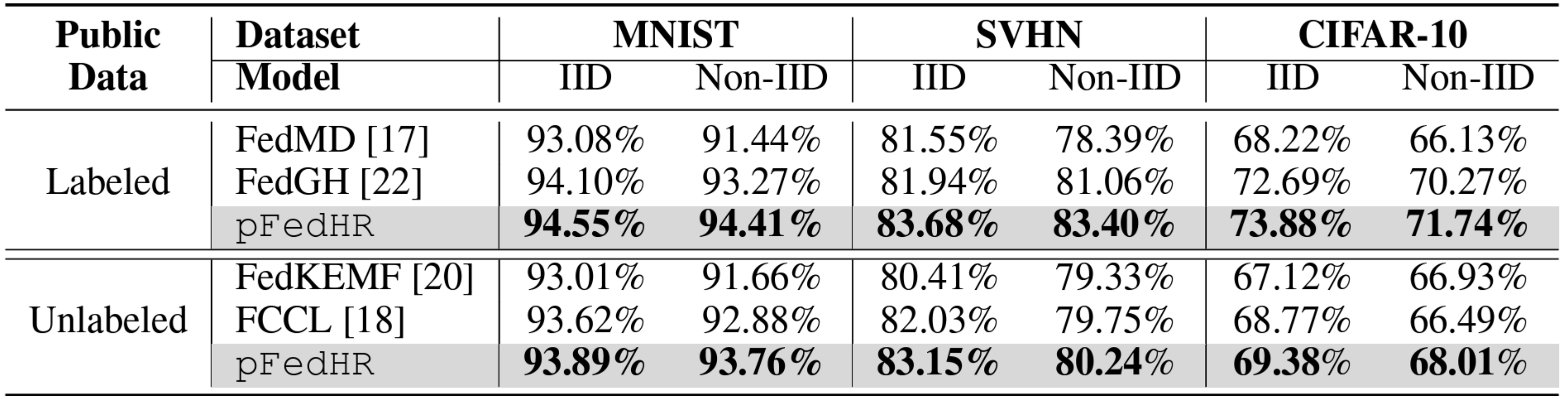


Wang et al. "Towards personalized federated learning via heterogeneous model reassembly." 37th Conference on Neural Information Processing Systems, 2023.

pFedHR

- Results

Table 2: Performance comparison with baselines under the heterogeneous setting.



Public Data	Dataset / Model	MNIST IID	MNIST Non-IID	SVHN IID	SVHN Non-IID	CIFAR-10 IID	CIFAR-10 Non-IID
Labeled	FedMD [17]	93.08%	91.44%	81.55%	78.39%	68.22%	66.13%
	FedGH [22]	94.10%	93.27%	81.94%	81.06%	72.69%	70.27%
	pFedHR	**94.55%**	**94.41%**	**83.68%**	**83.40%**	**73.88%**	**71.74%**
Unlabeled	FedKEMF [20]	93.01%	91.66%	80.41%	79.33%	67.12%	66.93%
	FCCL [18]	93.62%	92.88%	82.03%	79.75%	68.77%	66.49%
	pFedHR	**93.89%**	**93.76%**	**83.15%**	**80.24%**	**69.38%**	**68.01%**

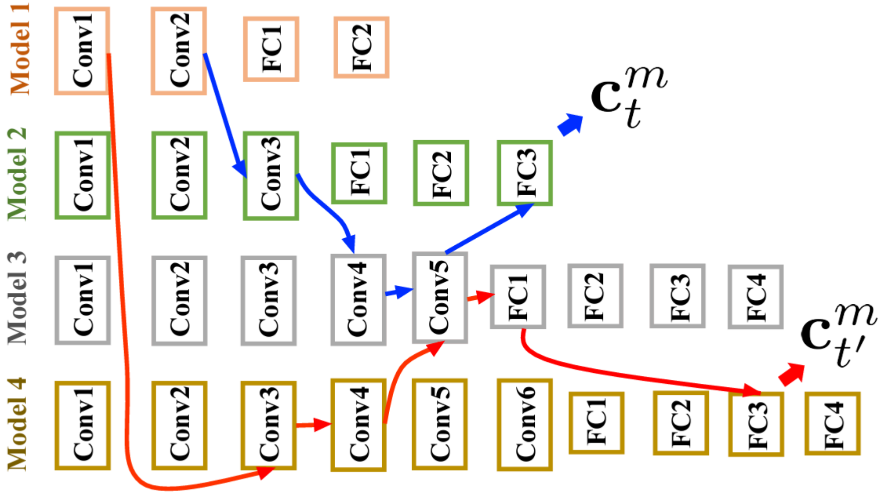
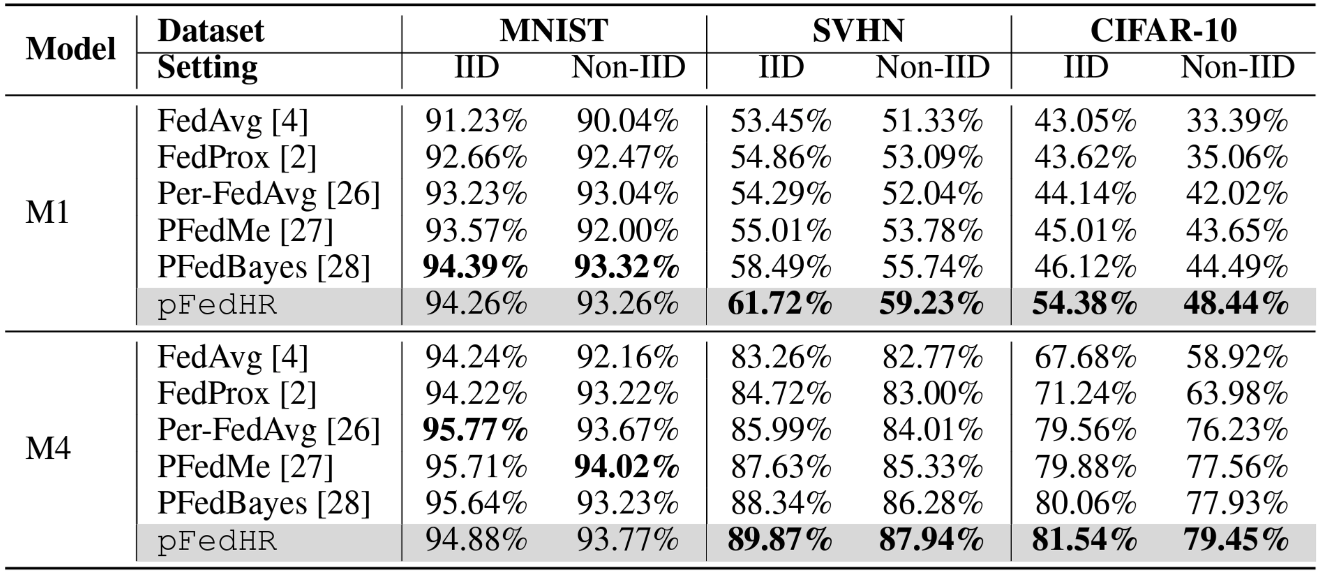


Table 4: Homogeneous model comparison with baselines.



Model	Dataset / Setting	MNIST IID	MNIST Non-IID	SVHN IID	SVHN Non-IID	CIFAR-10 IID	CIFAR-10 Non-IID
M1	FedAvg [4]	91.23%	90.04%	53.45%	51.33%	43.05%	33.39%
	FedProx [2]	92.66%	92.47%	54.86%	53.09%	43.62%	35.06%
	Per-FedAvg [26]	93.23%	93.04%	54.29%	52.04%	44.14%	42.02%
	PFedMe [27]	93.57%	92.00%	55.01%	53.78%	45.01%	43.65%
	PFedBayes [28]	**94.39%**	**93.32%**	58.49%	55.74%	46.12%	44.49%
	pFedHR	94.26%	93.26%	**61.72%**	**59.23%**	**54.38%**	**48.44%**
M4	FedAvg [4]	94.24%	92.16%	83.26%	82.77%	67.68%	58.92%
	FedProx [2]	94.22%	93.22%	84.72%	83.00%	71.24%	63.98%
	Per-FedAvg [26]	**95.77%**	93.67%	85.99%	84.01%	79.56%	76.23%
	PFedMe [27]	95.71%	**94.02%**	87.63%	85.33%	79.88%	77.56%
	PFedBayes [28]	95.64%	93.23%	88.34%	86.28%	80.06%	77.93%
	pFedHR	94.88%	93.77%	**89.87%**	**87.94%**	**81.54%**	**79.45%**

PennState

Wang et al. “Towards personalized federated learning via heterogeneous model reassembly.” 37th Conference on Neural Information Processing Systems, 2023.

FedType

Public Data Usage

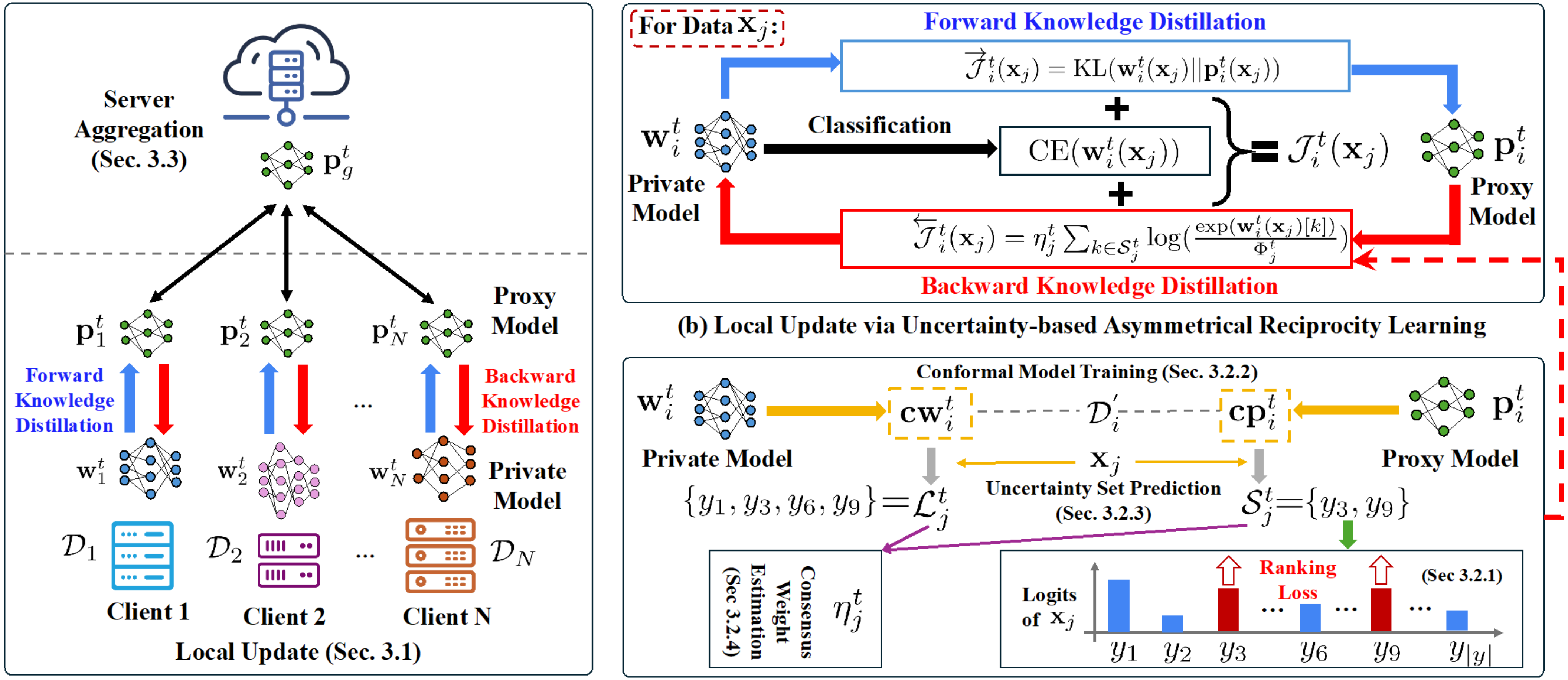
Sensitive Information Exchange

Communication Efficiency

PennState

Wang et al. “Bridging Model Heterogeneity in Federated Learning via Uncertainty-based Asymmetrical Reciprocity Learning.” under review, 2024.

FedType



(b) Local Update via Uncertainty-based Asymmetrical Reciprocity Learning

(a) Overview of FedType

(c) Uncertainty-based Behavior Imitation Learning (Sec. 3.2)

Wang et al. “Bridging Model Heterogeneity in Federated Learning via Uncertainty-based Asymmetrical Reciprocity Learning.” under review, 2024.

FedType

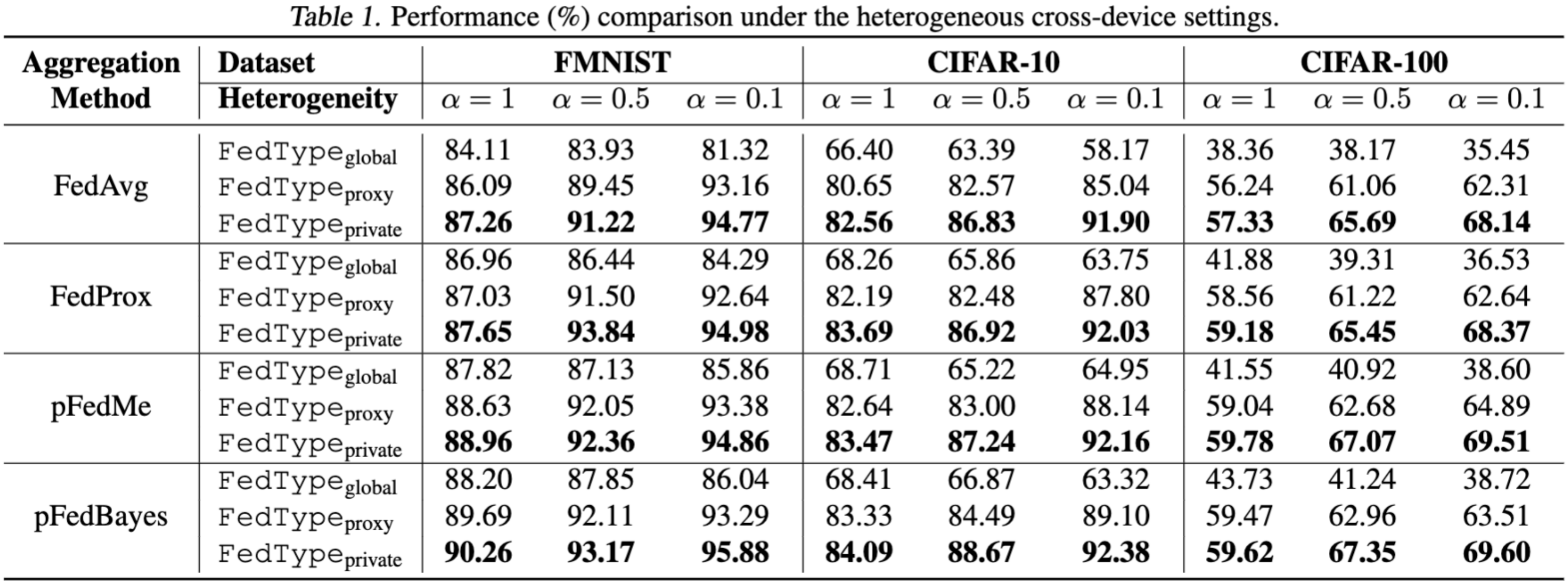
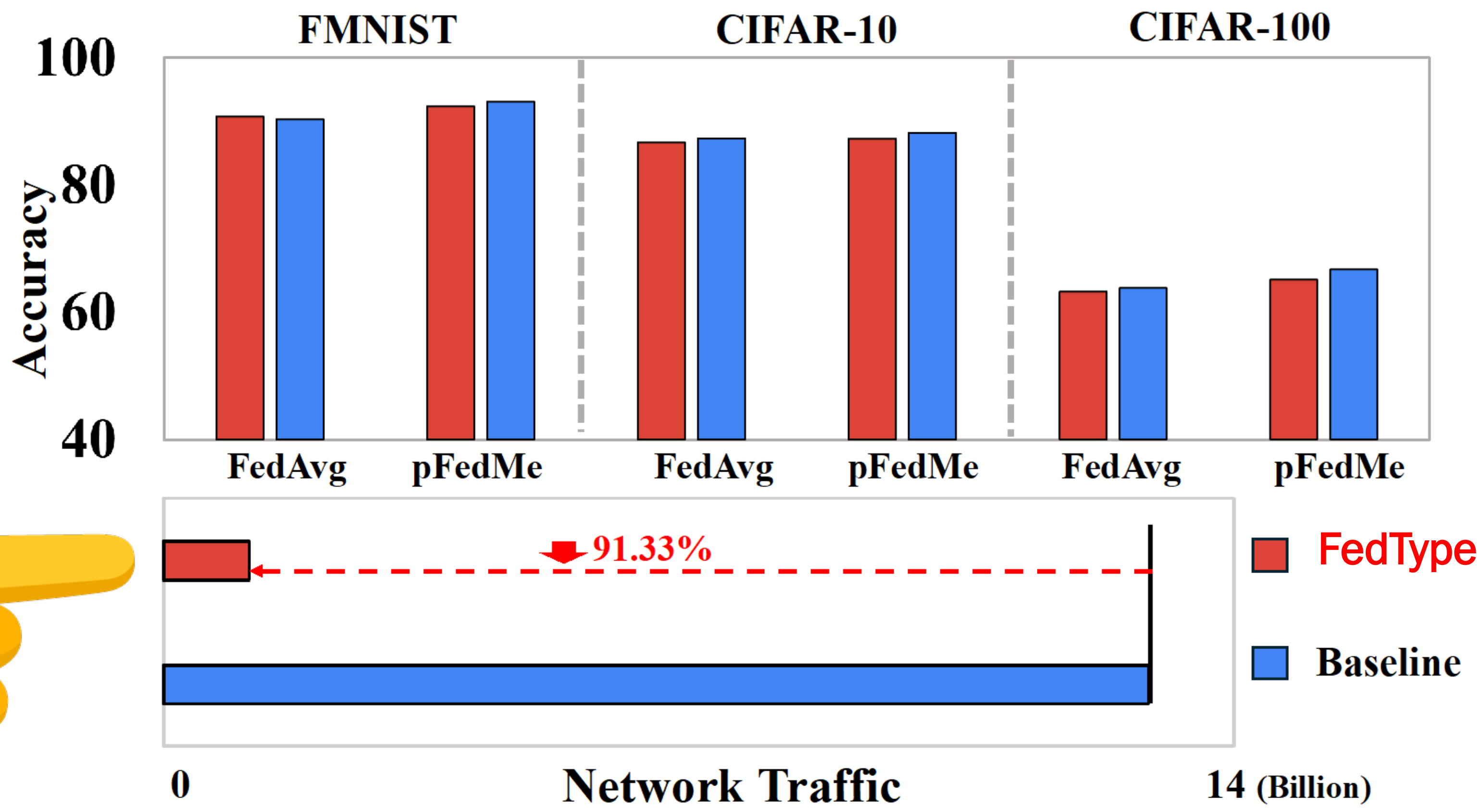


Table 1. Performance (%) comparison under the heterogeneous cross-device settings.

Aggregation Method	**Dataset**	**FMNIST**			**CIFAR-10**			**CIFAR-100**		
	Heterogeneity	$\alpha = 1$	$\alpha = 0.5$	$\alpha = 0.1$	$\alpha = 1$	$\alpha = 0.5$	$\alpha = 0.1$	$\alpha = 1$	$\alpha = 0.5$	$\alpha = 0.1$
FedAvg	$\texttt{FedType}_{\text{global}}$	84.11	83.93	81.32	66.40	63.39	58.17	38.36	38.17	35.45
	$\texttt{FedType}_{\text{proxy}}$	86.09	89.45	93.16	80.65	82.57	85.04	56.24	61.06	62.31
	$\texttt{FedType}_{\text{private}}$	**87.26**	**91.22**	**94.77**	**82.56**	**86.83**	**91.90**	**57.33**	**65.69**	**68.14**
FedProx	$\texttt{FedType}_{\text{global}}$	86.96	86.44	84.29	68.26	65.86	63.75	41.88	39.31	36.53
	$\texttt{FedType}_{\text{proxy}}$	87.03	91.50	92.64	82.19	82.48	87.80	58.56	61.22	62.64
	$\texttt{FedType}_{\text{private}}$	**87.65**	**93.84**	**94.98**	**83.69**	**86.92**	**92.03**	**59.18**	**65.45**	**68.37**
pFedMe	$\texttt{FedType}_{\text{global}}$	87.82	87.13	85.86	68.71	65.22	64.95	41.55	40.92	38.60
	$\texttt{FedType}_{\text{proxy}}$	88.63	92.05	93.38	82.64	83.00	88.14	59.04	62.68	64.89
	$\texttt{FedType}_{\text{private}}$	**88.96**	**92.36**	**94.86**	**83.47**	**87.24**	**92.16**	**59.78**	**67.07**	**69.51**
pFedBayes	$\texttt{FedType}_{\text{global}}$	88.20	87.85	86.04	68.41	66.87	63.32	43.73	41.24	38.72
	$\texttt{FedType}_{\text{proxy}}$	89.69	92.11	93.29	83.33	84.49	89.10	59.47	62.96	63.51
	$\texttt{FedType}_{\text{private}}$	**90.26**	**93.17**	**95.88**	**84.09**	**88.67**	**92.38**	**59.62**	**67.35**	**69.60**

Wang et al. “Bridging Model Heterogeneity in Federated Learning via Uncertainty-based Asymmetrical Reciprocity Learning.” under review, 2024.

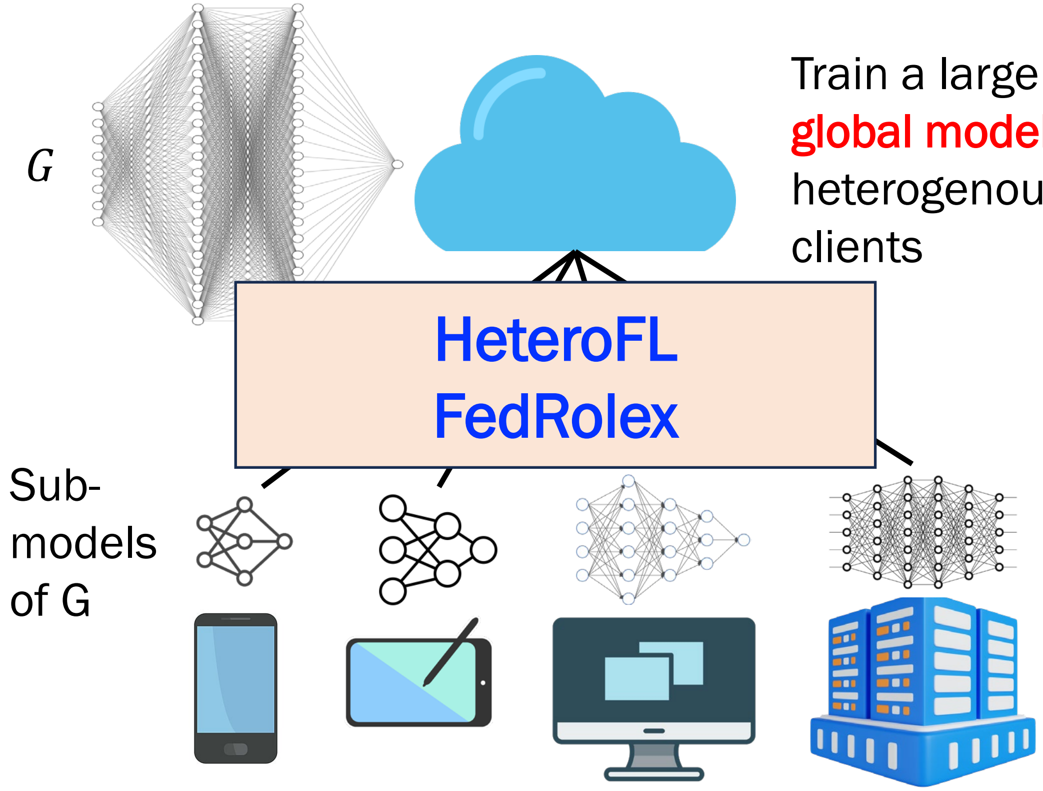
FedType



Communication efficiency analysis

Wang et al. “Bridging Model Heterogeneity in Federated Learning via Uncertainty-based Asymmetrical Reciprocity Learning.” under review, 2024.

Model Heterogeneity

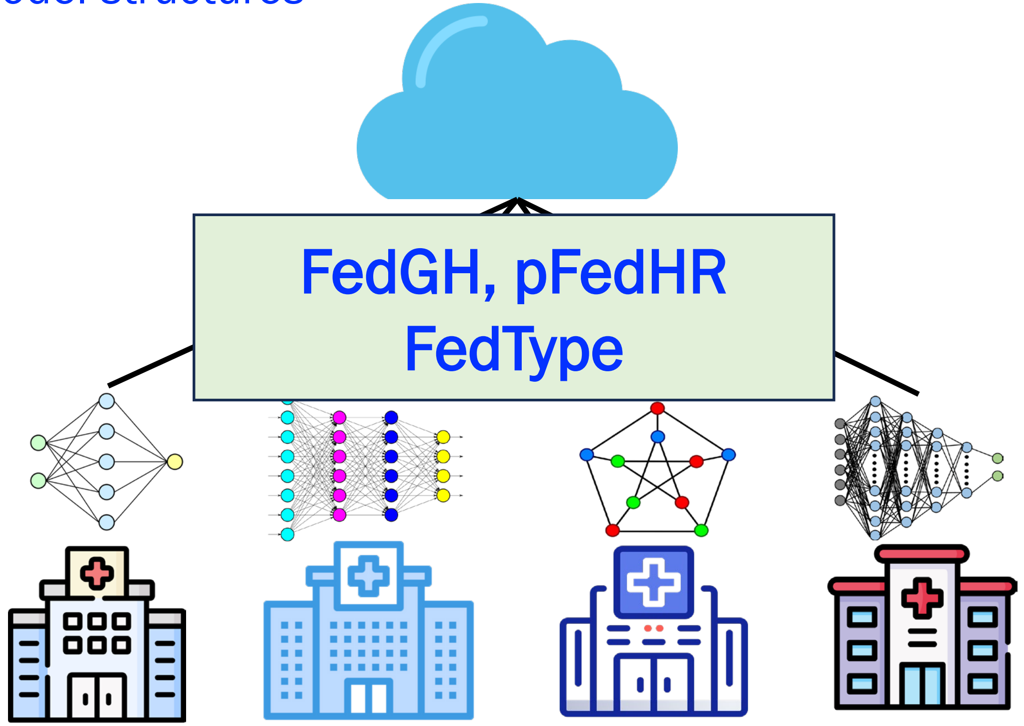


HeteroFL
FedRolex

Train a large **global model** with heterogenous clients

Sub-model training (partial heterogeneity)

Enhance the performance of each **client model** through collaborative learning without modifying client model structures



FedGH, pFedHR
FedType

Heterogeneous model aggregation (complete heterogeneity)

Part 4

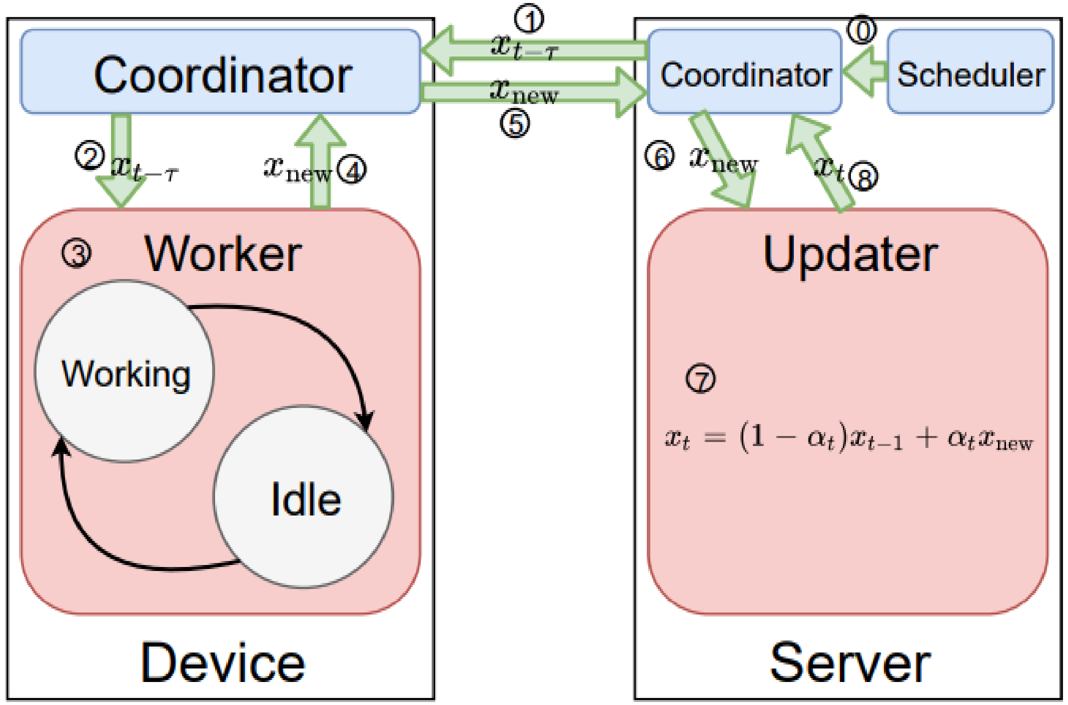
- Part 1: Federated Learning Introduction
- Part 2: Data/Statistical Heterogeneity
- Part 3: Model Heterogeneity
- Part 4: System Heterogeneity
- Part 5: Conclusion and Future Work

FedAsync

- Motivation
  - Different clients may have different capabilities to process and communicate.
  - When handling massive edge devices, there could be a large number of stragglers. The synchronous mechanism could be slow.

Xie et al. “Asynchronous Federated Optimization.” 12th Annual Workshop on Optimization for Machine Learning (OPT). 2020.

FedAsync

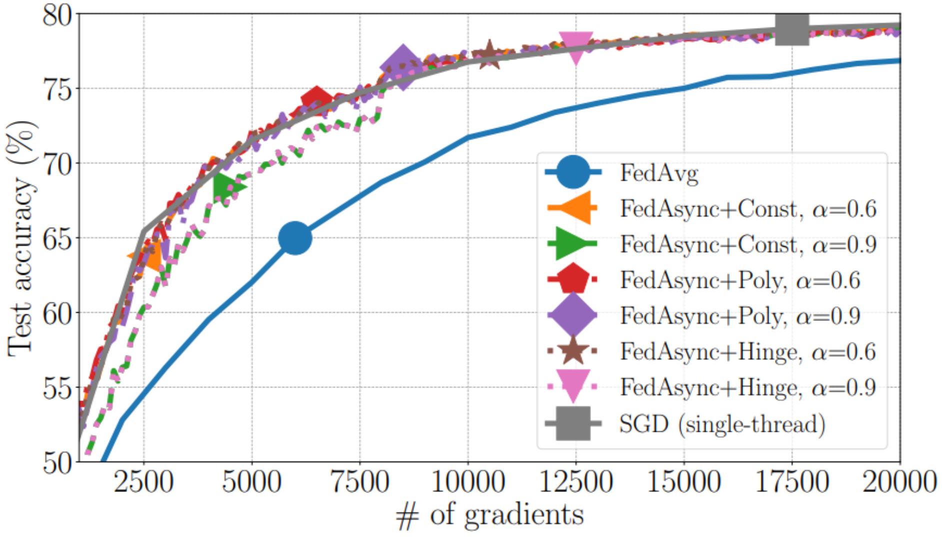


- Step 0: scheduler triggers training through coordinator
- Step 1-2: worker receives model $x_{t-\tau}$ from server via coordinator
- Step 3: worker computes local updates
- Step 4-6: worker pushes the locally updated model to server via the coordinator. Coordinator queues the models received in 5, and feeds them to the updater sequentially in 6
- Step 7-8: server updates the global model and makes it ready to read in the coordinator
- Step 1 and 5 operate asynchronously in parallel

Xie et al. “Asynchronous Federated Optimization.” 12th Annual Workshop on Optimization for Machine Learning (OPT). 2020.

FedAsync

- Selected Results



(a) Top-1 accuracy on testing set, $t - \tau \leq 4$

CIFAR-10 dataset. Alpha is the hyperparameter, cons, poly, and hinge are different weighting functions to decide alpha_t.

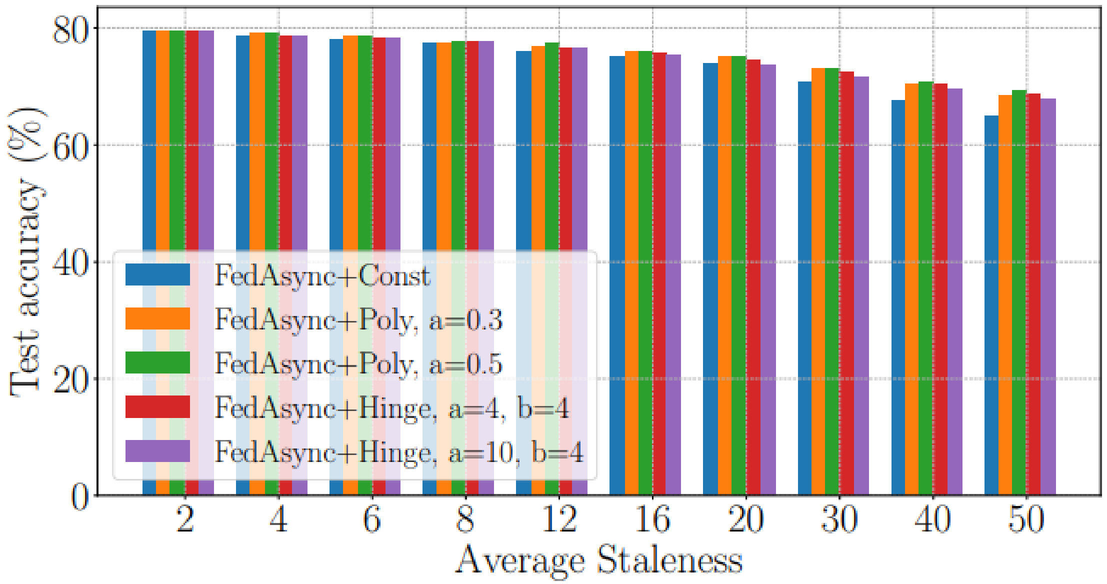


Figure 4: Top-1 accuracy on CNN and CIFAR-10 dataset at the end of training, with different staleness. $\gamma = 0.1$, $\rho = 0.01$. α has initial value 0.9.

Learning rate , Regularization weights

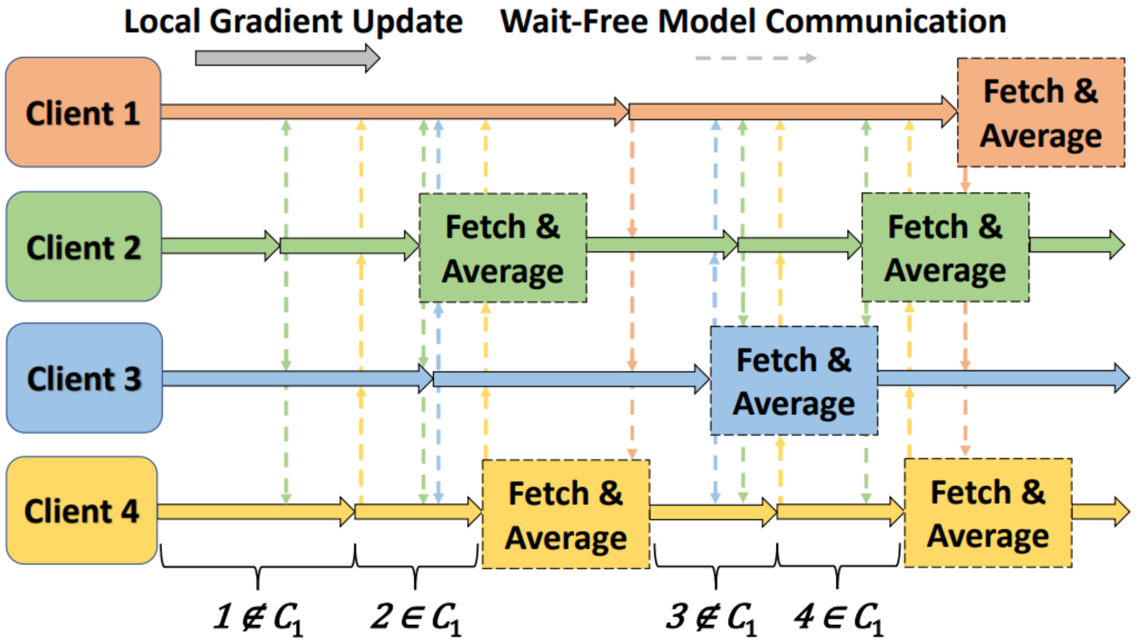
Xie et al. "Asynchronous Federated Optimization." 12th Annual Workshop on Optimization for Machine Learning (OPT). 2020.

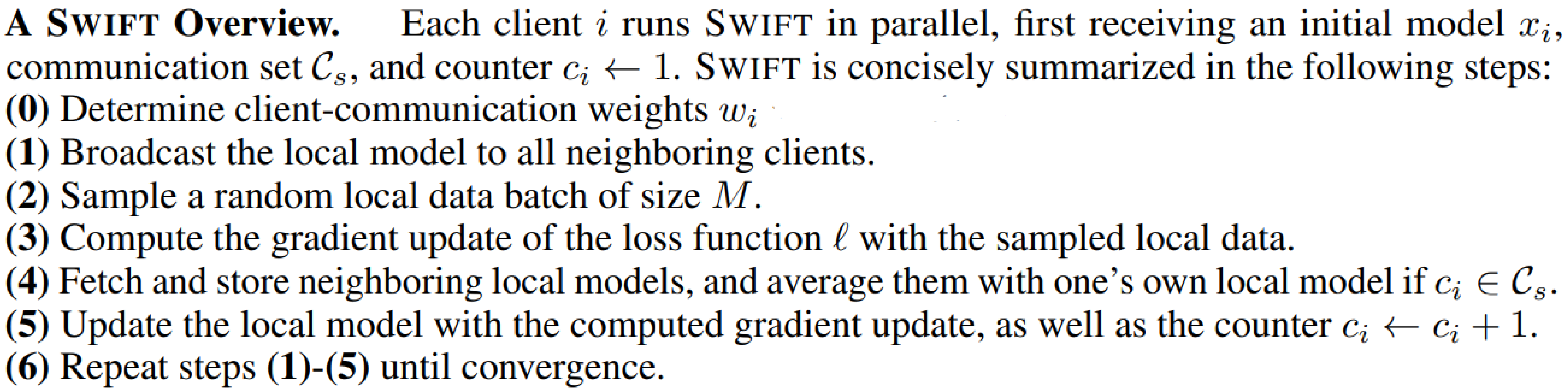
SWIFT

- Motivation:
  - Synchronous nature of current decentralized FL algorithms, communication time per round, and consequently run-time, is amplified by parallelization delays. These delays are caused by the slowest client in the network.
  - Some exiting research work either do not propagate models well throughout the network (via gossip algorithms) or require partial synchronization.
  - These asynchronous algorithms rely on a deterministic bounded-delay assumption, which ensures that the slowest client in the network updates at least every τ iterations. This assumption is strong and worsen the convergence.
- Contribution: a novel wait-free decentralized FL algorithm that allows clients to conduct training at their own speed.

Bornstein et al. “SWIFT: Rapid Decentralized Federated Learning via Wait-Free Model Communication.” The Eleventh International Conference on Learning Representations (ICLR), 2023.

SWIFT

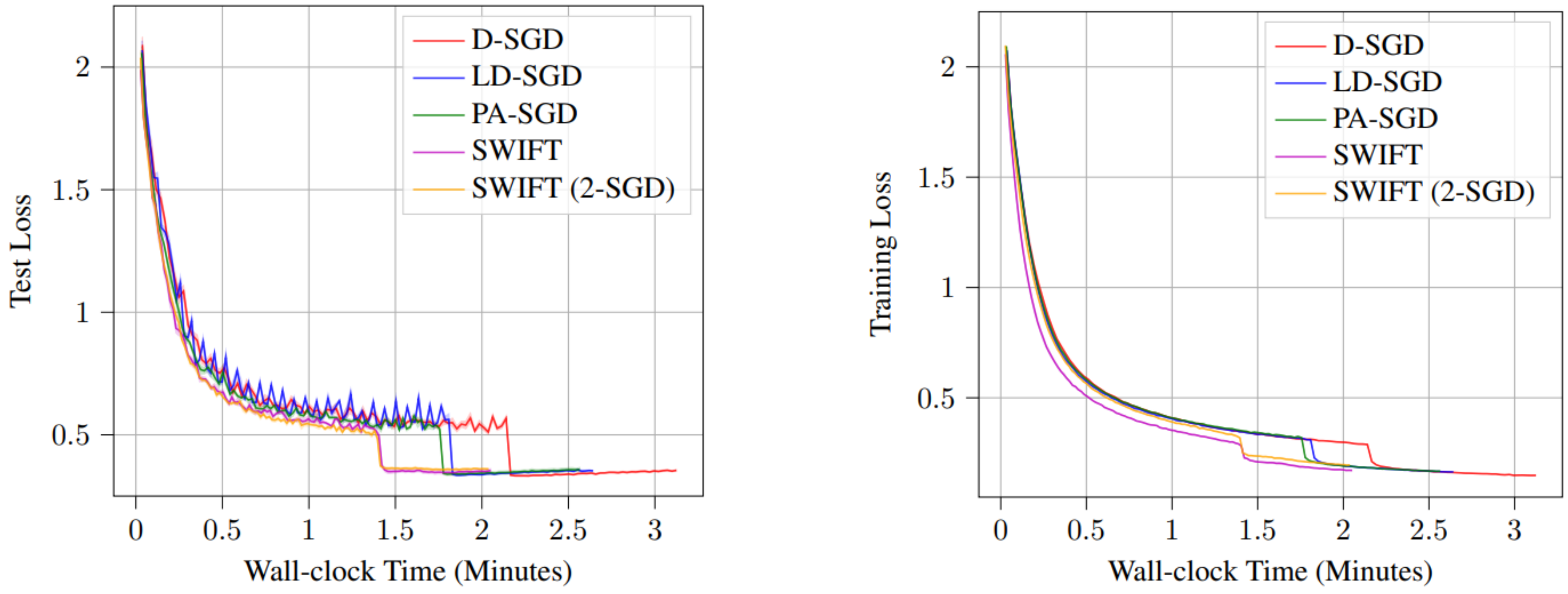



A SWIFT Overview. Each client i runs SWIFT in parallel, first receiving an initial model x_i, communication set $\mathcal{C}_s$, and counter $c_i \leftarrow 1$. SWIFT is concisely summarized in the following steps:
(0) Determine client-communication weights w_i
(1) Broadcast the local model to all neighboring clients.
(2) Sample a random local data batch of size M.
(3) Compute the gradient update of the loss function ℓ with the sampled local data.
(4) Fetch and store neighboring local models, and average them with one's own local model if $c_i \in \mathcal{C}_s$.
(5) Update the local model with the computed gradient update, as well as the counter $c_i \leftarrow c_i + 1$.
(6) Repeat steps **(1)**-**(5)** until convergence.

Bornstein et al. "SWIFT: Rapid Decentralized Federated Learning via Wait-Free Model Communication." The Eleventh International Conference on Learning Representations (ICLR), 2023.

SWIFT

- Results



(a) Average test loss.

(b) Average train loss.

Figure 2: Baseline performance comparison on CIFAR-10 for 16 client ring.

Bornstein et al. “SWIFT: Rapid Decentralized Federated Learning via Wait-Free Model Communication.” The Eleventh International Conference on Learning Representations (ICLR), 2023.

Part 5

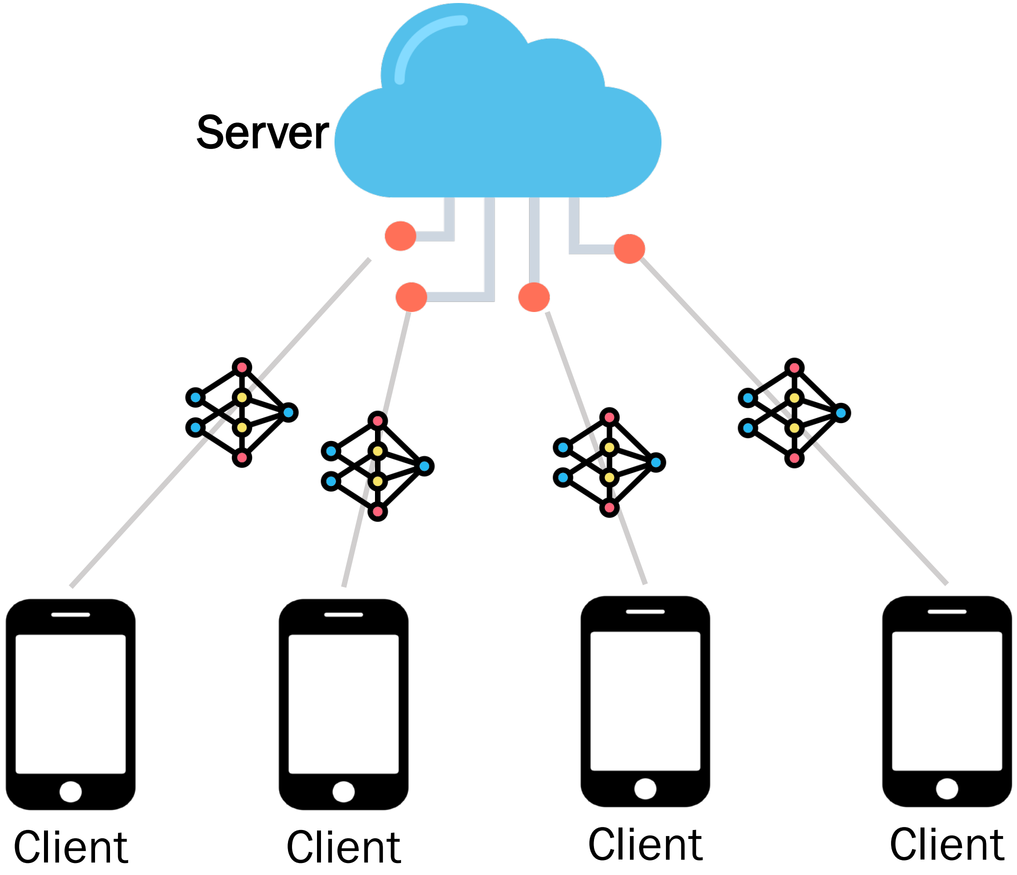
- Part 1: Federated Learning Introduction
- Part 2: Data/Statistical Heterogeneity
- Part 3: Model Heterogeneity
- Part 4: System Heterogeneity
- Part 5: Conclusion and Future Work

Core Challenges of Federated Learning

- Communication Efficiency

- Privacy Concerns

- Heterogeneity
  - Data/Statistical Heterogeneity
  - Model Heterogeneity
  - System Heterogeneity



Multimodal Federated Learning

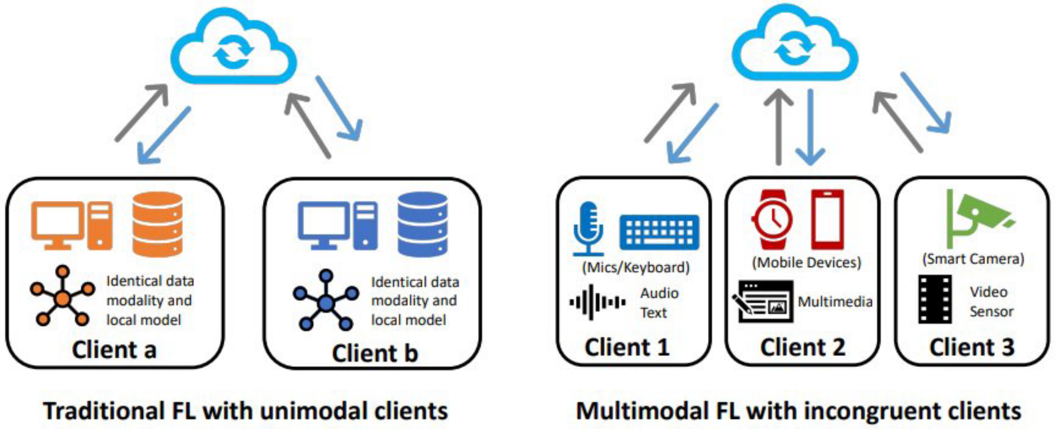

Identical data modality and local model
Client a
Identical data modality and local model
Client b
(Mics/Keyboard)
Audio
Text
Client 1
(Mobile Devices)
Multimedia
Client 2
(Smart Camera)
Video
Sensor
Client 3
Traditional FL with unimodal clients
Multimodal FL with incongruent clients

Figure 1. Illustration of traditional unimodal FL v.s. multimodal FL.

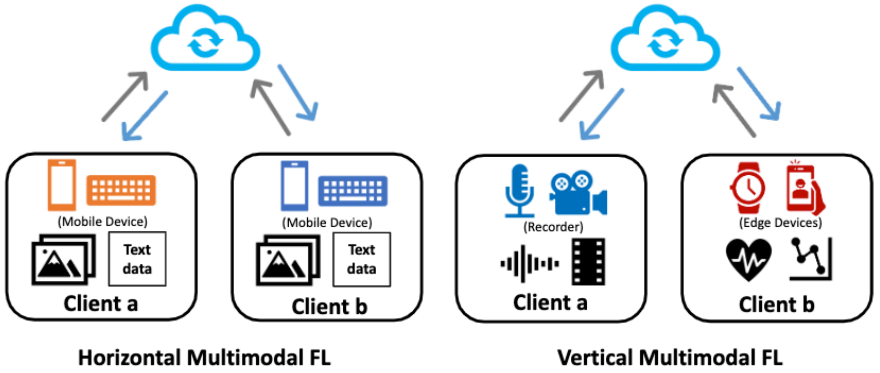


Figure 4. The illustration of Horizontal Multimodal Federated Learning and Vertical Multimodal Federated Learning. *Left:* Horizontal Multimodal Federated Learning contains two clients. Both hold image and text data. *Right:* Vertical Multimodal Federated Learning example contains two clients with exclusive modalities. Client *a* has audio and video data, while client *b* holds heat rate and acceleration sensor data.

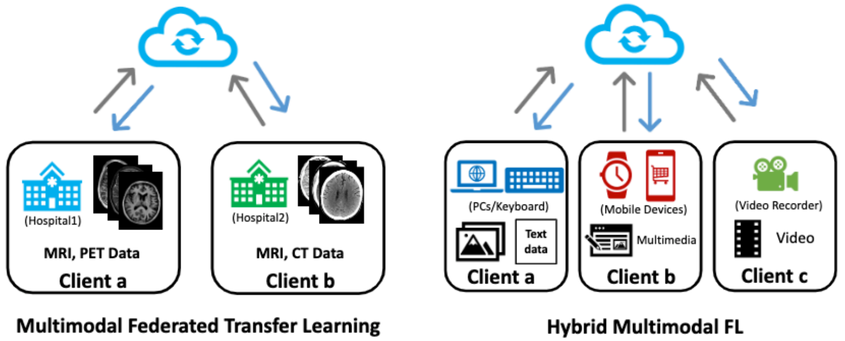


Figure 5. The illustration of Multimodal Federated Transfer Learning and Hybrid Multimodal Federated Learning. *Left:* Multimodal Federated Transfer Learning contains two hospitals as clients. One holds MRI and PET data, the other holds MRI and CT data. *Right:* Hybrid Multimodal Federated Learning example contains three clients with different modality combinations. The system contains both unimodal and multimodal clients.

Che et al. "Multimodal federated learning: A survey." Sensors 23.15 (2023): 6986.

Fedmultimodal

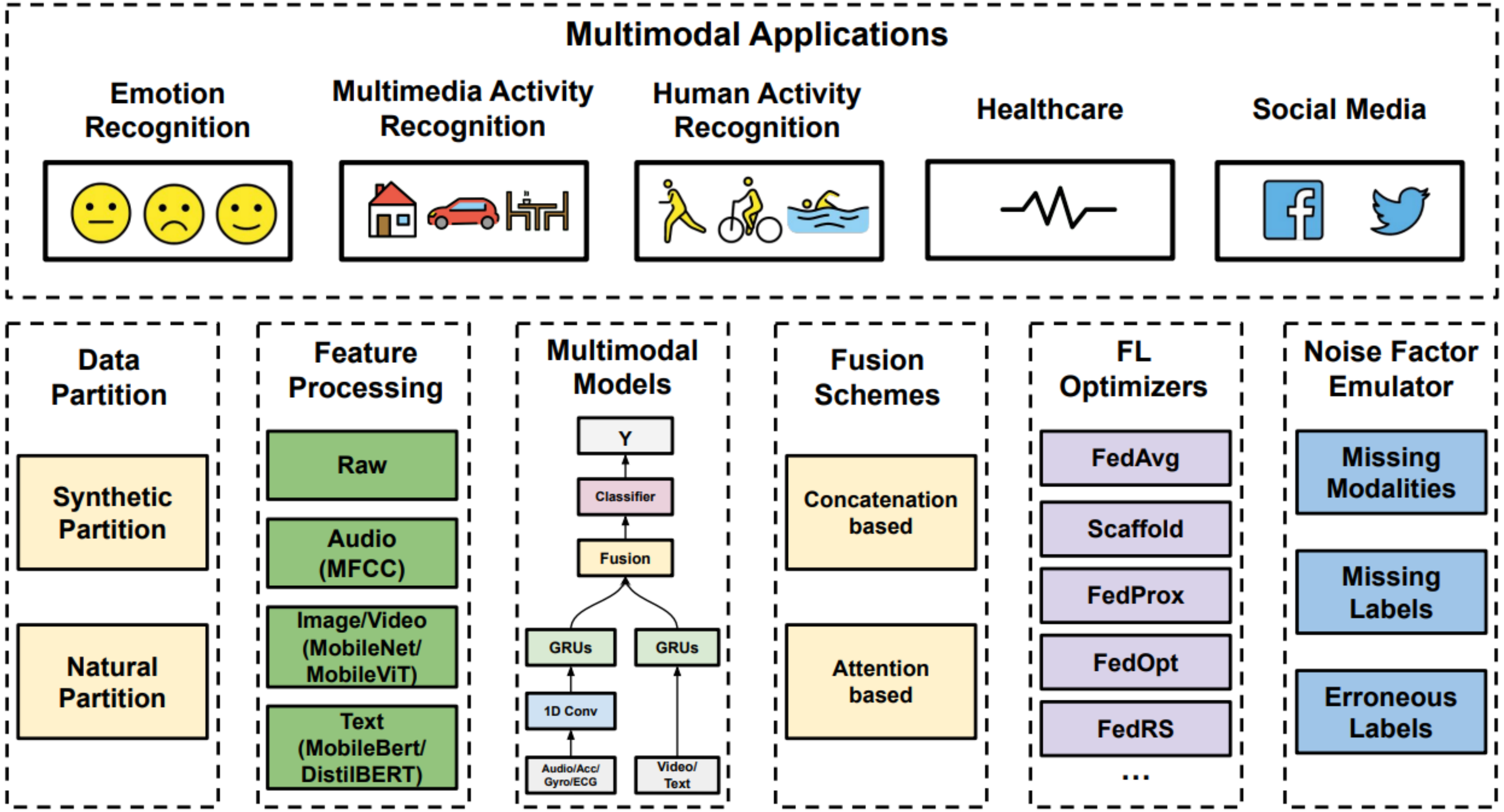
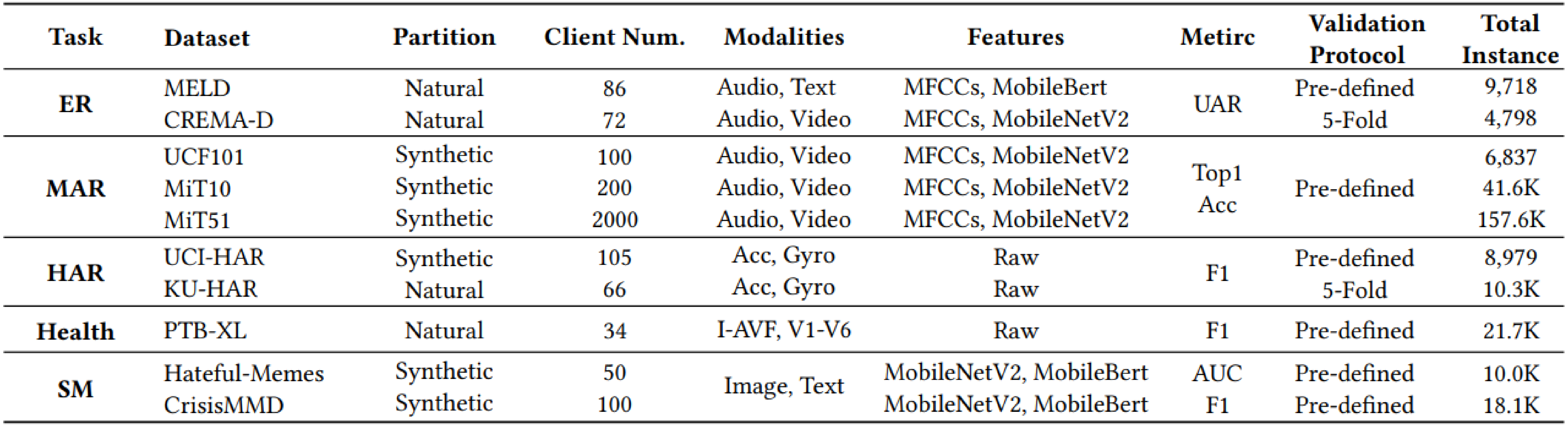


Figure 1: The overall architecture of the end-to-end multimodal federated learning framework included in FedMultimodal.

Feng et al. "Fedmultimodal: A benchmark for multimodal federated learning." Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD). 2023.

Fedmultimodal

Table 2: Overview of the 10 datasets included in FedMultimodal.



Task	Dataset	Partition	Client Num.	Modalities	Features	Metirc	Validation Protocol	Total Instance
ER	MELD	Natural	86	Audio, Text	MFCCs, MobileBert	UAR	Pre-defined	9,718
	CREMA-D	Natural	72	Audio, Video	MFCCs, MobileNetV2		5-Fold	4,798
MAR	UCF101	Synthetic	100	Audio, Video	MFCCs, MobileNetV2	Top1 Acc	Pre-defined	6,837
	MiT10	Synthetic	200	Audio, Video	MFCCs, MobileNetV2			41.6K
	MiT51	Synthetic	2000	Audio, Video	MFCCs, MobileNetV2			157.6K
HAR	UCI-HAR	Synthetic	105	Acc, Gyro	Raw	F1	Pre-defined	8,979
	KU-HAR	Natural	66	Acc, Gyro	Raw		5-Fold	10.3K
Health	PTB-XL	Natural	34	I-AVF, V1-V6	Raw	F1	Pre-defined	21.7K
SM	Hateful-Memes	Synthetic	50	Image, Text	MobileNetV2, MobileBert	AUC	Pre-defined	10.0K
	CrisisMMD	Synthetic	100		MobileNetV2, MobileBert	F1	Pre-defined	18.1K

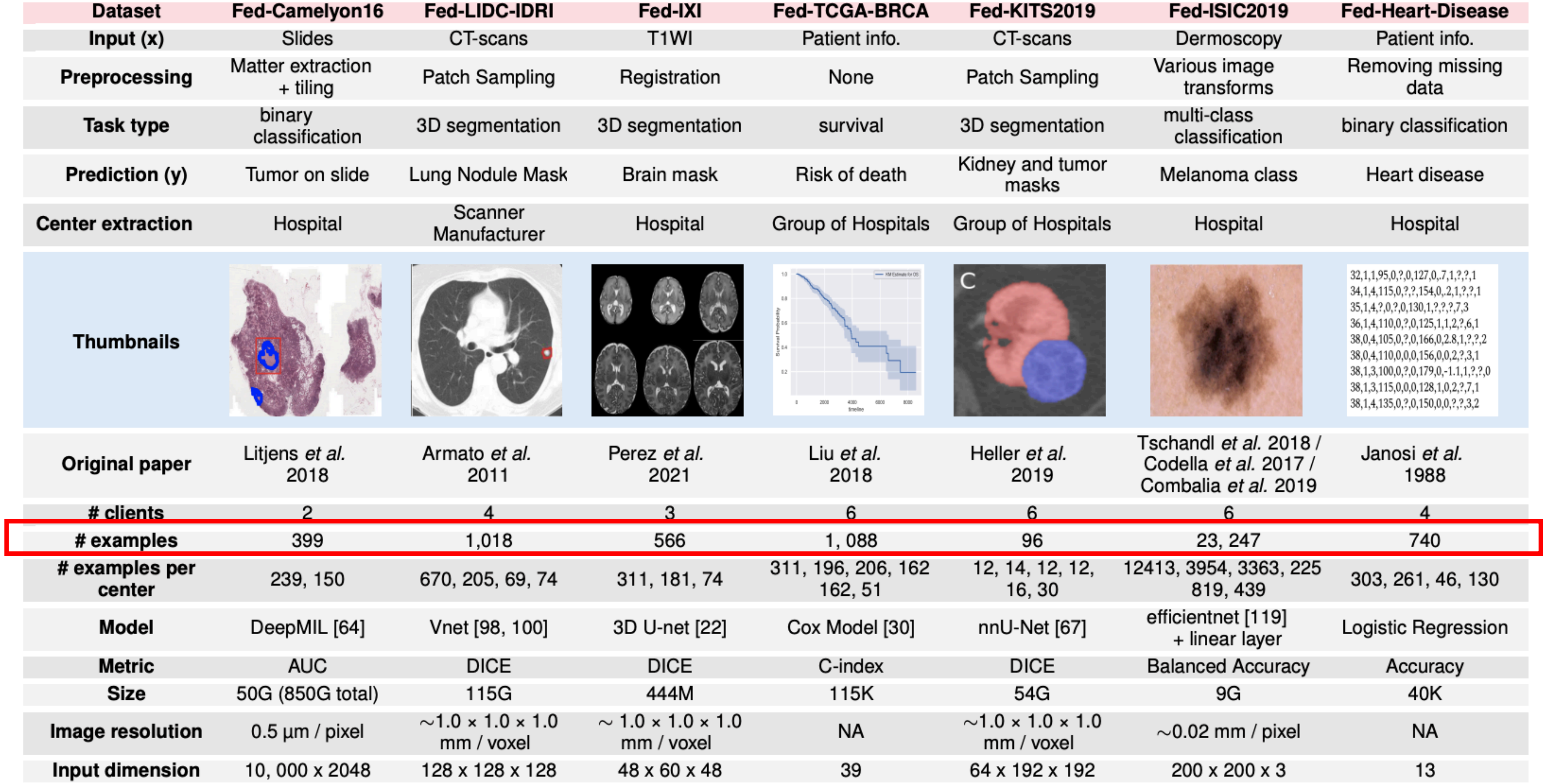
PennState

Feng et al. "Fedmultimodal: A benchmark for multimodal federated learning." Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD). 2023.

Domain-specific Federated Learning Systems

- Healthcare
  - FLamby

How to train FL models with limited number of data?



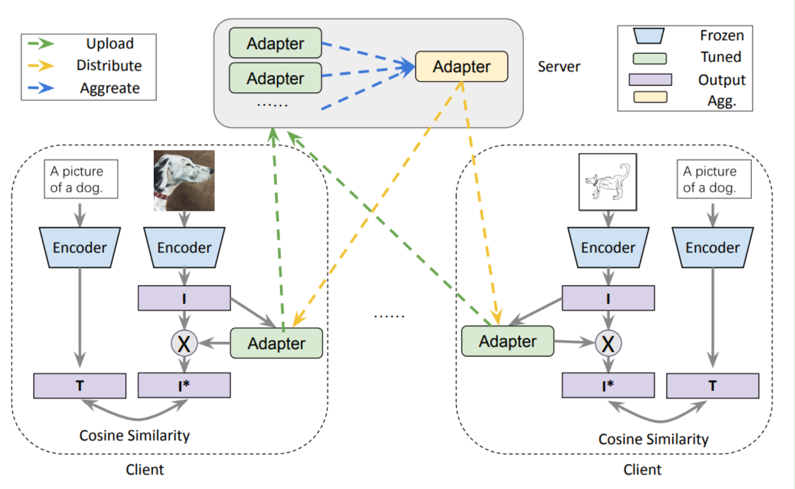
Dataset	Fed-Camelyon16	Fed-LIDC-IDRI	Fed-IXI	Fed-TCGA-BRCA	Fed-KITS2019	Fed-ISIC2019	Fed-Heart-Disease
Input (x)	Slides	CT-scans	T1WI	Patient info.	CT-scans	Dermoscopy	Patient info.
Preprocessing	Matter extraction + tiling	Patch Sampling	Registration	None	Patch Sampling	Various image transforms	Removing missing data
Task type	binary classification	3D segmentation	3D segmentation	survival	3D segmentation	multi-class classification	binary classification
Prediction (y)	Tumor on slide	Lung Nodule Mask	Brain mask	Risk of death	Kidney and tumor masks	Melanoma class	Heart disease
Center extraction	Hospital	Scanner Manufacturer	Hospital	Group of Hospitals	Group of Hospitals	Hospital	Hospital
Thumbnails							32,1,1,95,0,?,0,127,0,.7,1,?,?,1 34,1,4,115,0,?,?,154,0,.2,1,?,?,1 35,1,4,?,0,?,0,130,1,?,?,?,7,3 36,1,4,110,0,?,0,125,1,1,2,?,6,1 38,0,4,105,0,?,0,166,0,2.8,1,?,?,2 38,0,4,110,0,0,0,156,0,0,2,?,3,1 38,1,3,100,0,?,0,179,0,-1.1,1,?,?,0 38,1,3,115,0,0,0,128,1,0,2,?,7,1 38,1,4,135,0,?,0,150,0,0,?,?,3,2
Original paper	Litjens *et al.* 2018	Armato *et al.* 2011	Perez *et al.* 2021	Liu *et al.* 2018	Heller *et al.* 2019	Tschandl *et al.* 2018 / Codella *et al.* 2017 / Combalia *et al.* 2019	Janosi *et al.* 1988
# clients	2	4	3	6	6	6	4
# examples	399	1,018	566	1, 088	96	23, 247	740
# examples per center	239, 150	670, 205, 69, 74	311, 181, 74	311, 196, 206, 162 162, 51	12, 14, 12, 12, 16, 30	12413, 3954, 3363, 225 819, 439	303, 261, 46, 130
Model	DeepMIL [64]	Vnet [98, 100]	3D U-net [22]	Cox Model [30]	nnU-Net [67]	efficientnet [119] + linear layer	Logistic Regression
Metric	AUC	DICE	DICE	C-index	DICE	Balanced Accuracy	Accuracy
Size	50G (850G total)	115G	444M	115K	54G	9G	40K
Image resolution	0.5 μm / pixel	~1.0 × 1.0 × 1.0 mm / voxel	~ 1.0 × 1.0 × 1.0 mm / voxel	NA	~1.0 × 1.0 × 1.0 mm / voxel	~0.02 mm / pixel	NA
Input dimension	10, 000 x 2048	128 x 128 x 128	48 x 60 x 48	39	64 x 192 x 192	200 x 200 x 3	13

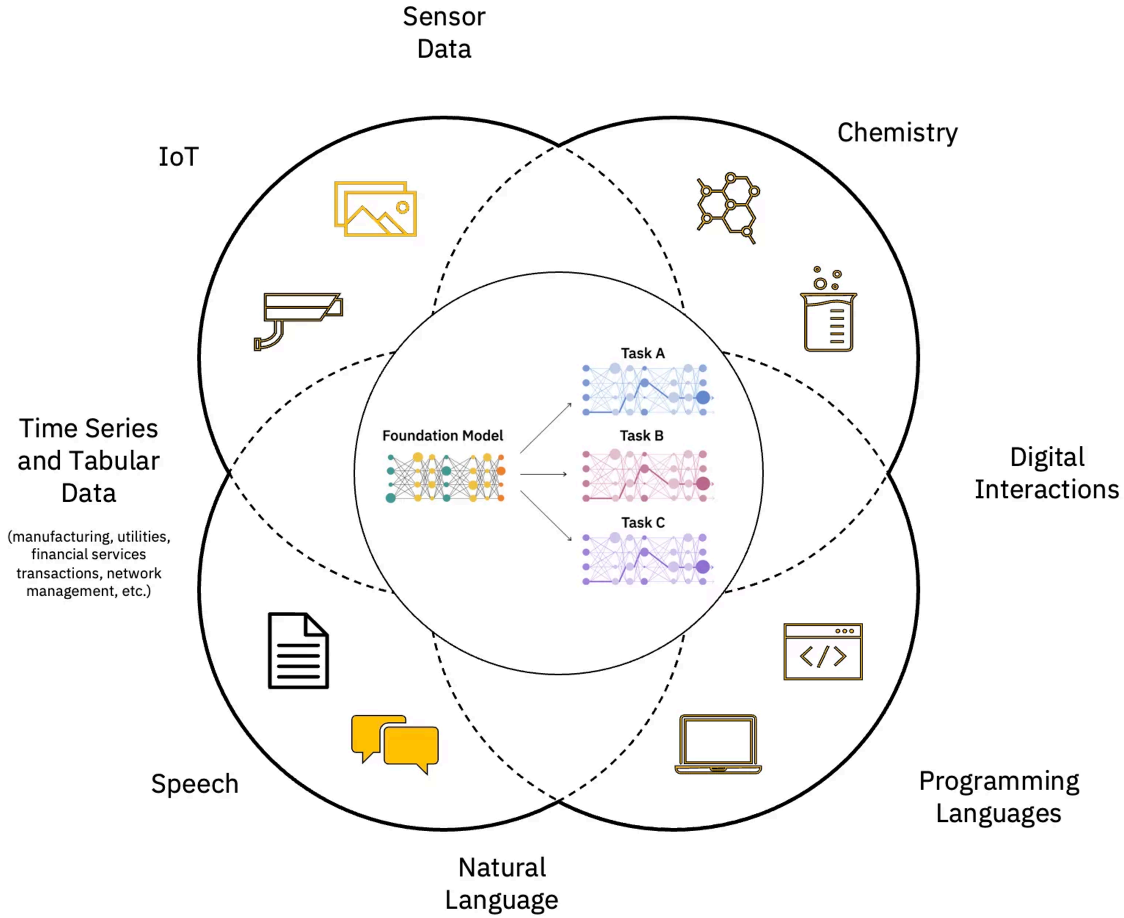
Terrail et al. "FLamby: Datasets and Benchmarks for Cross-Silo Federated Learning in Realistic Healthcare Settings." Advances in Neural Information Processing Systems, 2022.

Foundation Models + Federated Learning

- How to use foundation models to enhance client learning?
- Can we train a foundation model with federated learning?

FedCLIP





PennState

Zhuang et al. “When Foundation Model Meets Federated Learning: Motivations, Challenges, and Future Directions.” arXiv:2306.15546, 2023.
Lu et al. "FedCLIP: Fast Generalization and Personalization for CLIP in Federated Learning." IEEE Data Engineering Bulletin 2023.

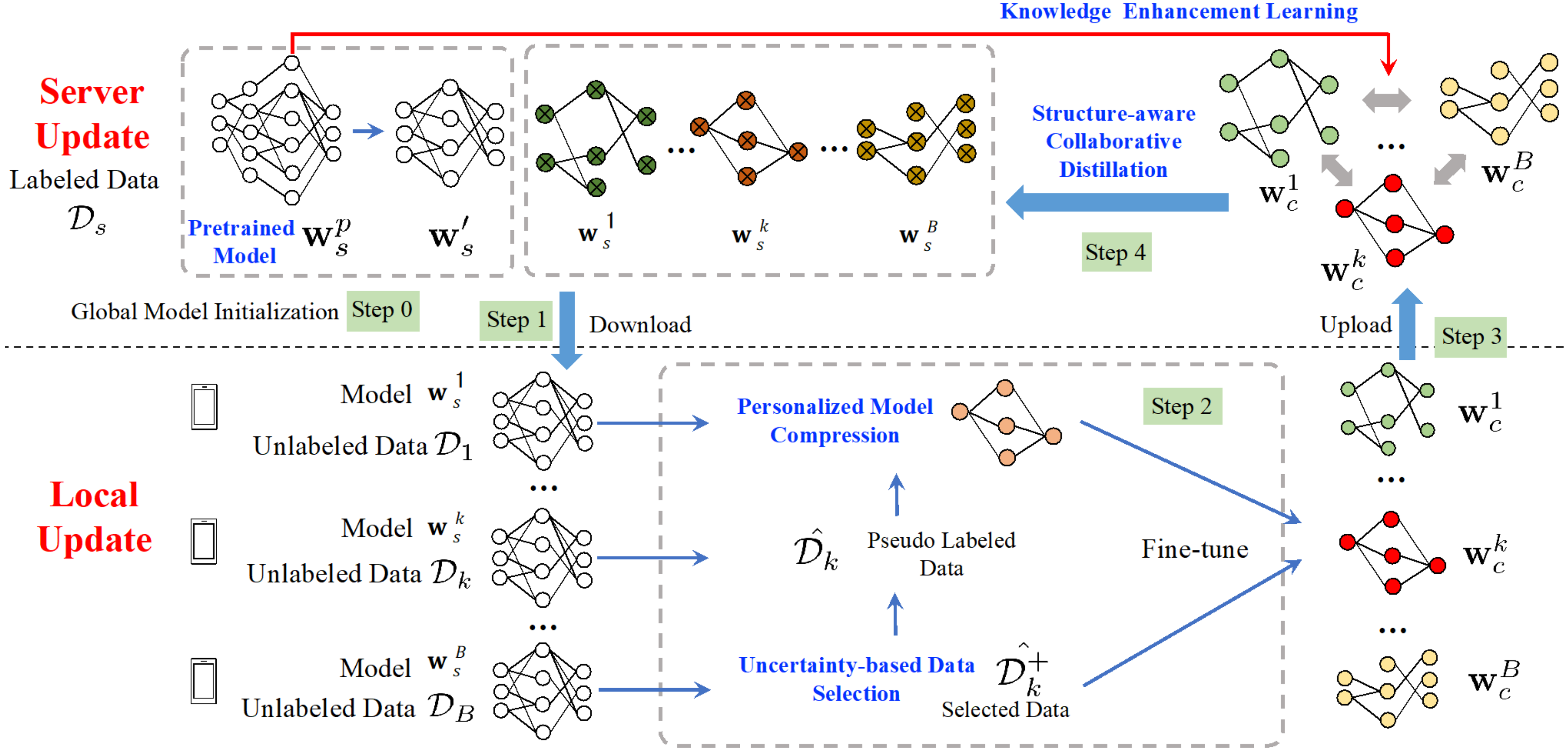
Other Federated Learning Settings

Supervised

Semi/weakly-Supervised

Unsupervised

pFedKnow (Semi-supervised FL)



Wang et al. "Knowledge-enhanced semi-supervised federated learning for aggregating heterogeneous lightweight clients in IoT." Proceedings of the 2023 SIAM International Conference on Data Mining (SDM). 2023.

Thank You.

Any questions, please feel free contact Jiaqi Wang or Fenglong Ma
via jqwang@psu.edu or fenglong@psu.edu