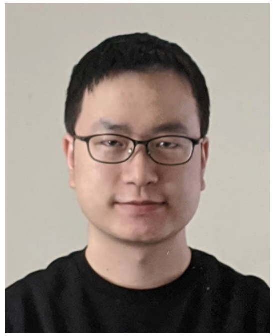
AIFARMS
Artificial Intelligence for Future Agricultural Resilience, Management, and Sustainability

Personalized Federated Learning with Parameter Propagation



Jun Wu[1]
junwu3@illinois.edu

Wenxuan Bao[1]
wbao3@illinois.edu

Elizabeth Ainsworth[1,2]
ainsworth@illinois.edu

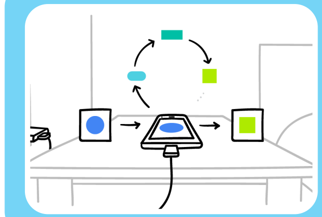
Jingrui He[1]
jingrui@illinois.edu

[1]University of Illinois at Urbana-Champaign
[2]USDA ARS Global Change and Photosynthesis Research Unit

ILLINOIS

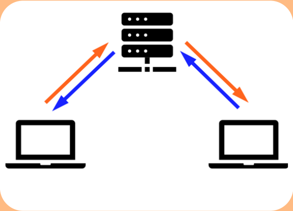
USDA Agricultural Research Service

Roadmap



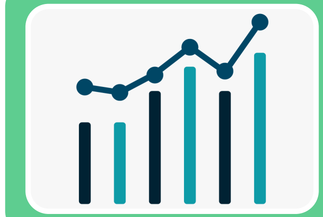
Background

- Personalized Federated Learning
- A Transfer Learning Perspective



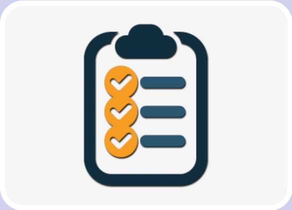
Methodology

- Federated Parameter Propagation
- Iterative Optimization



Experiments

- Performance Comparison
- Model Analysis



Conclusion

- Algorithm
- Evaluation

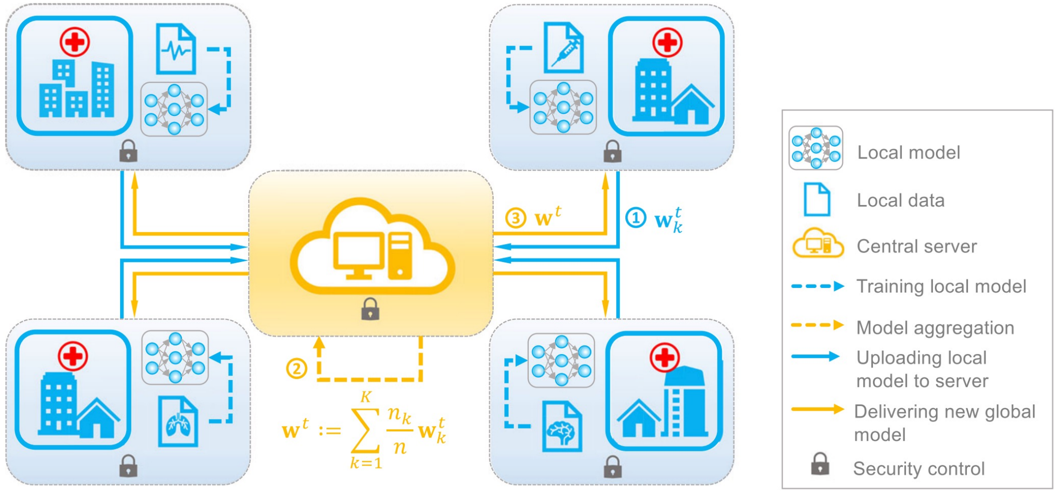
Federated Learning (FL)

❑ Definition

- Multiple clients **collaborate in solving a machine learning problem**, under the coordination of a central server or service provider.
- Each client's **raw data is stored locally** and not exchanged.

❑ Applications

(a) Mobile keyboard prediction



(b) Healthcare informatics

- Peter Kairouz, et al. "**Advances and open problems in federated learning**." Foundations and Trends® in Machine Learning 2021.
- Tian Li, et al. "**Federated learning: Challenges, methods, and future directions**." IEEE signal processing magazine 2020.
- Jie Xu, et al. "**Federated learning for healthcare informatics**." Journal of Healthcare Informatics Research. 2021

Federated Learning (FL)

❑ Workflow

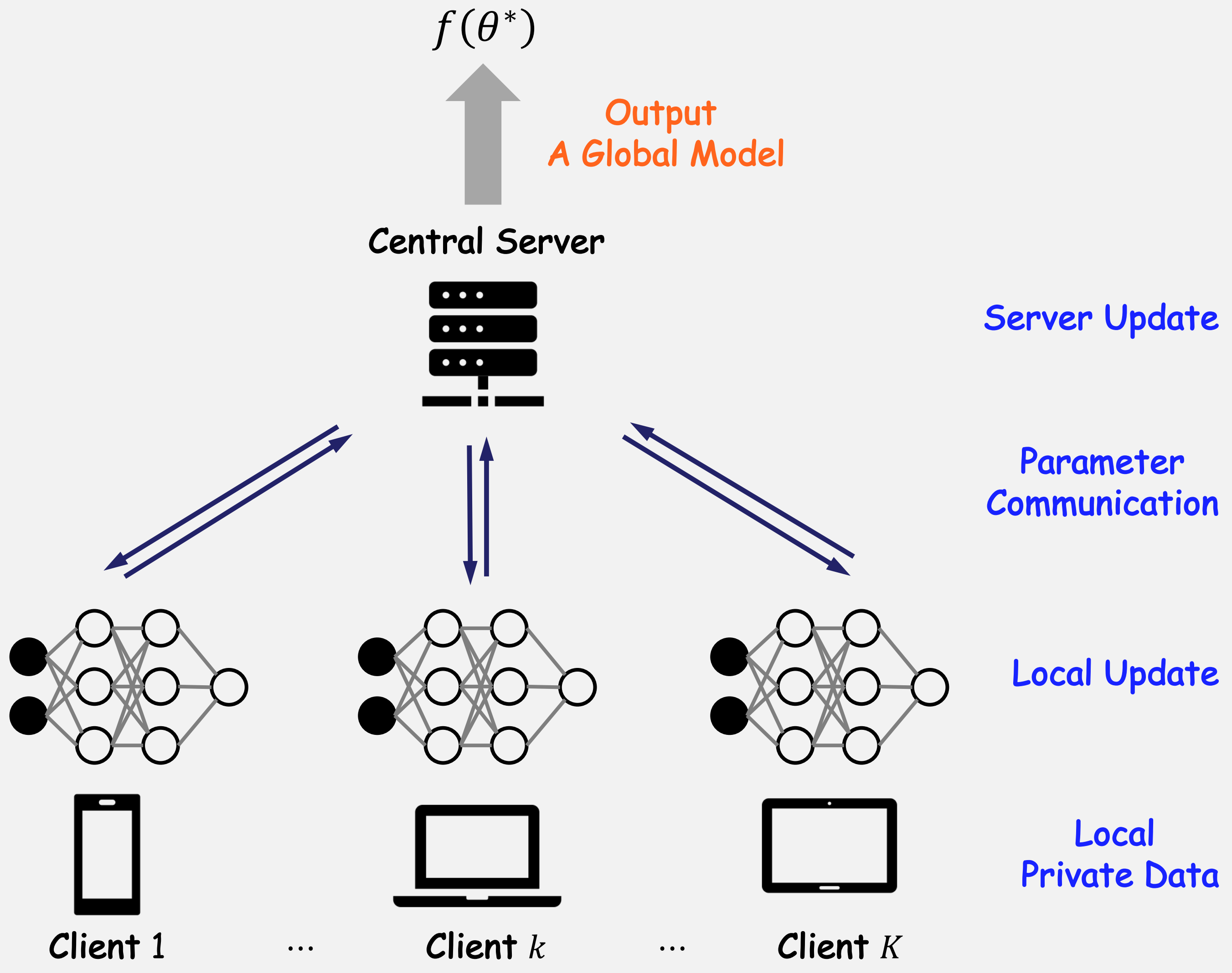
- **Client Update**: Locally update parameters w.r.t. private data

$$\theta_k \leftarrow \arg\min_{\theta} \ell(\theta; D_k)$$

- **Forward Communication**: Upload parameter updates to the server
- **Server Update**: Synchronously aggregate the received parameters

$$\theta_G \leftarrow \mathrm{AGG}(\theta_1, \theta_2, \cdots, \theta_K)$$

- **Backward Communication**: Sent the global parameters back to clients



- Brendan McMahan, et al . "**Communication-efficient learning of deep networks from decentralized data**." AISTATS 2017.
- Peter Kairouz, et al. "**Advances and open problems in federated learning**." Foundations and Trends® in Machine Learning 2021.

Personalized Federated Learning (pFL)

❑ Challenge

- Data heterogeneity

For some clients $i, j \in \{1,2,\cdots,K\}$:

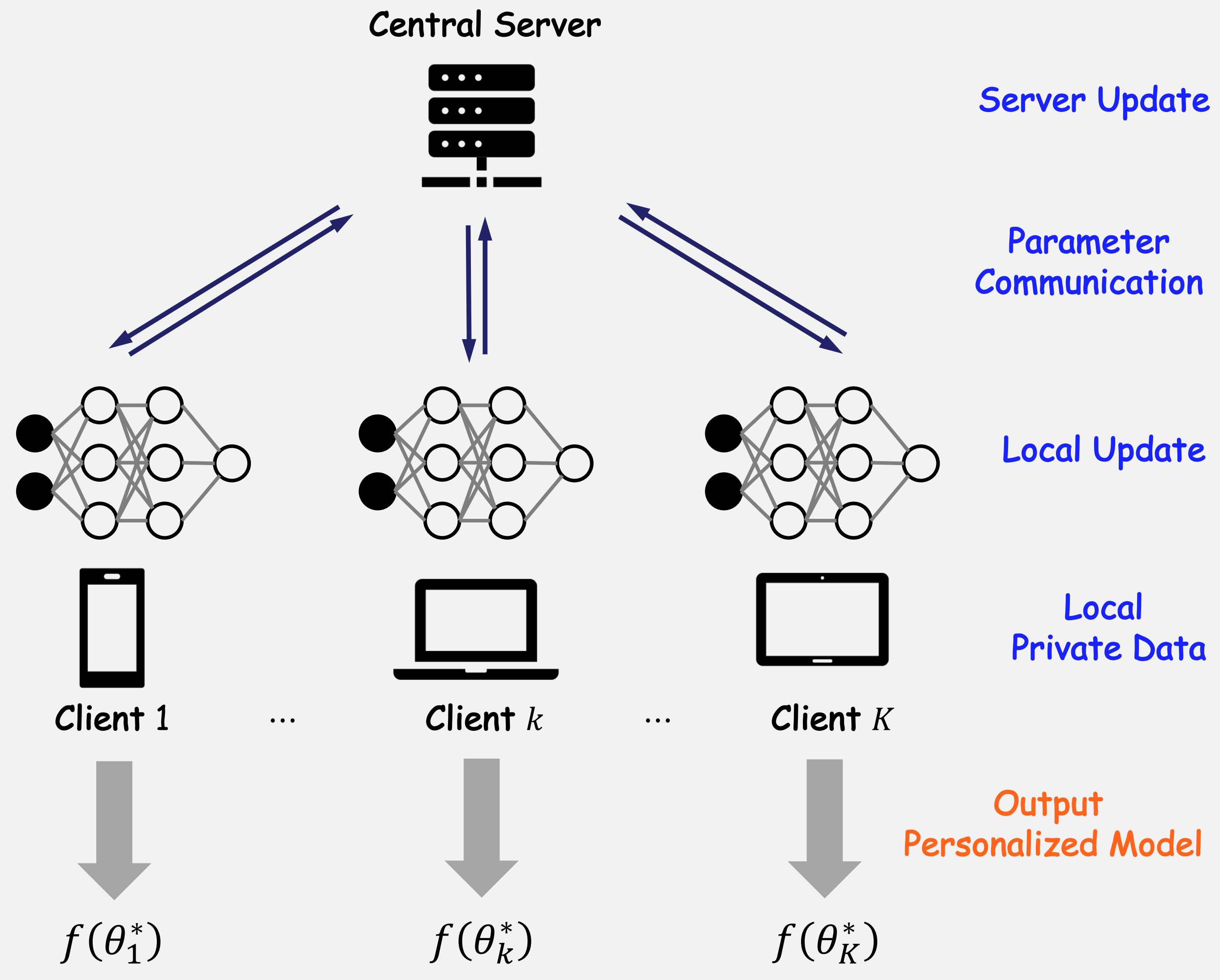
$$p_i(x,y) \neq p_j(x,y)$$

❑ Goal

- Learn a personalized model for each local client

For each client k:

$$\theta_k \leftarrow \arg\min_{\theta} \ell(\theta; D_k, \theta_1, \cdots, \theta_{k-1}, \theta_{k+1}, \cdots, \theta_K)$$

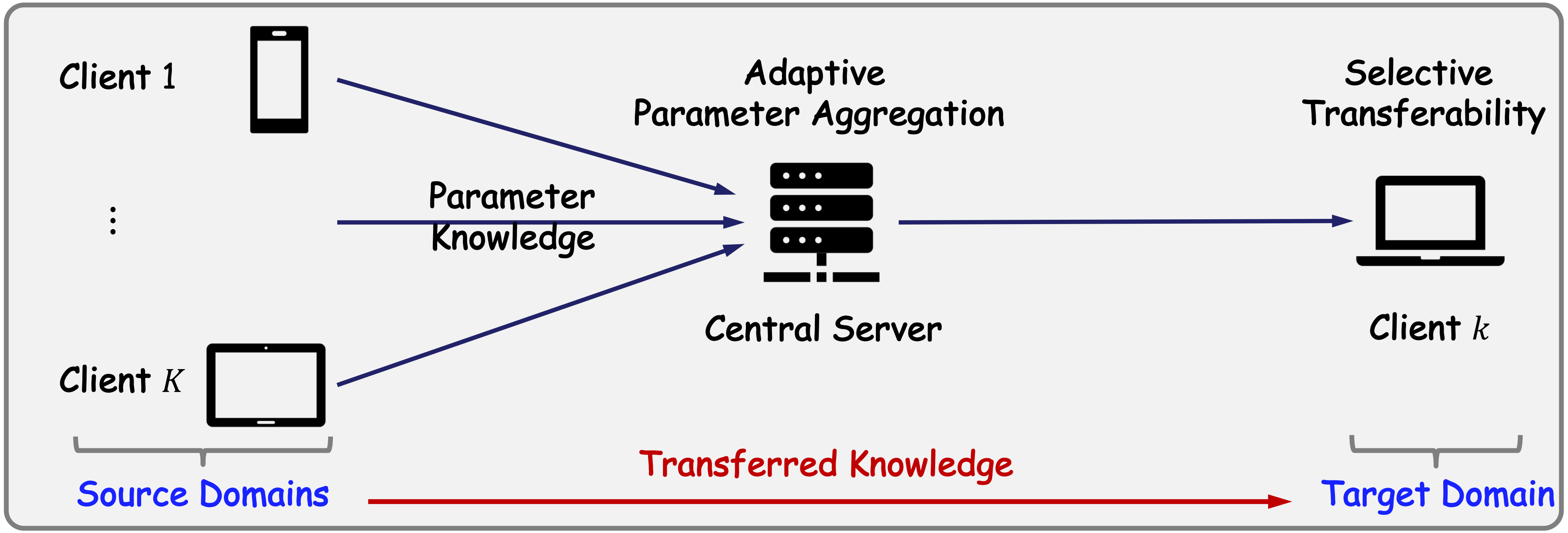


- Canh T Dinh, et al. "**Personalized federated learning with Moreau envelopes**." NeurIPS 2020.
- Aviv Shamsian, et al. "**Personalized federated learning using hypernetworks**." ICML 2021.

A Transfer Learning Perspective

❑ Knowledge Transfer across Clients

- Target domain: Any client $k \in \{1,2,\cdots,K\}$
- Source domains: All other clients $k' \neq k$
- Goal: For client k, it aims to **improve prediction performance** using source knowledge

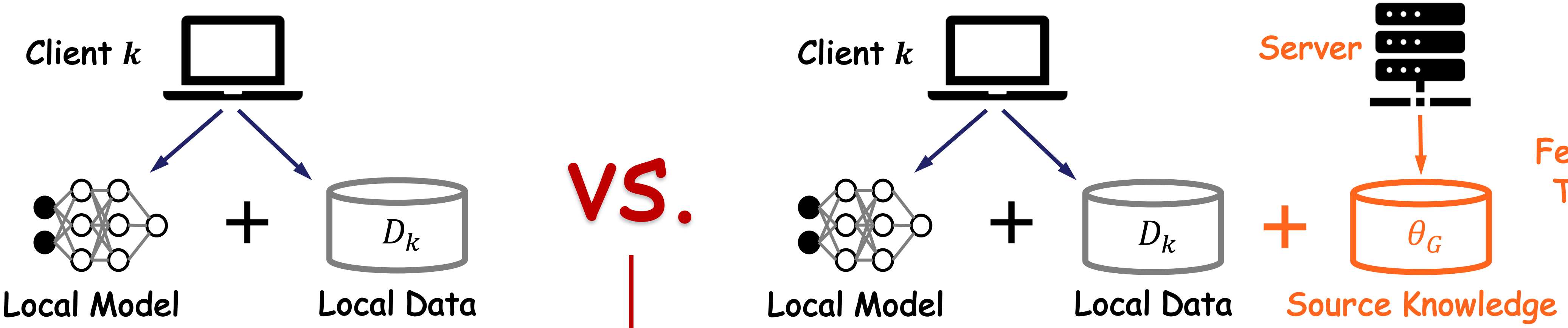


- Jun Wu, and Jingrui He. "**A unified meta-Learning framework for dynamic transfer learning.**" IJCAI 2022.

❑ Negative Transfer

- Transferring knowledge from the source can have a negative impact on the target learner

(a) Local Training



(b) Federated Training

LOCAL: $\theta^*_{LOCAL} = \arg\min_{\theta} \ell(\theta; D_k)$

pFL: $\theta^*_{pFL} = \arg\min_{\theta} \mathrm{pFL}(\theta; D_k, \theta_1, \cdots, \theta_K)$

(a) Local Training $\quad \mathrm{ACCURACY}(\theta^*_{LOCAL}; D_k^{test}) > \mathrm{ACCURACY}(\theta^*_{pFL}; D_k^{test}) \quad$ (b) Federated Training

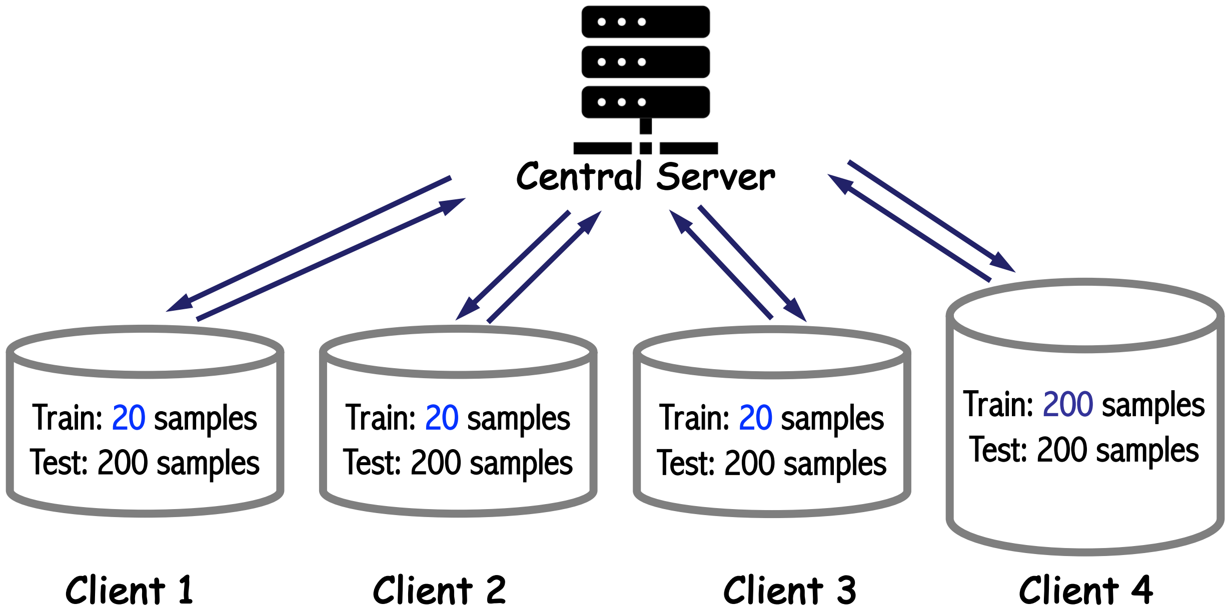
Transferred knowledge hurts generalization performance

- Zirui Wang, et al. "**Characterizing and avoiding negative transfer**." CVPR 2019.

Whether Negative Transfer Happens?

❑ Observations

- Existing pFL algorithms suffer from negative transfer
- Negative transfer is more likely to happen for client with adequate training samples



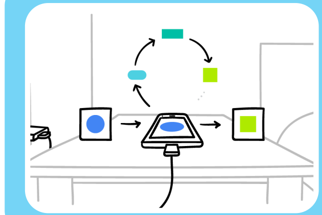
(a) Imbalanced training samples across clients

Model	Accuracy				Average Accuracy
	Client 1	Client 2	Client 3	Client 4	
LOCAL	**0.5270**	**0.4840**	**0.4980**	**0.8110**	0.5800
FedAvg	**0.3755**	**0.4420**	0.6455	**0.7965**	0.5649
LG-FedAvg	0.5440	0.5115	0.5430	**0.8095**	0.6020
Ditto	**0.4095**	**0.4810**	0.6465	**0.8095**	0.5866
FedAMP	0.5300	0.5210	0.5415	**0.8105**	0.6008

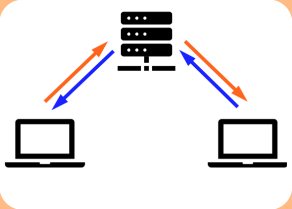
(b) Results of personalized federated learning

- Zirui Wang, et al. "**Characterizing and avoiding negative transfer**." CVPR 2019.

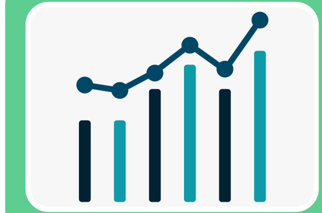
Roadmap



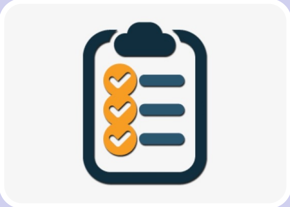
Background
- Personalized Federated Learning
- A Transfer Learning Perspective



Methodology
- Federated Parameter Propagation
- Iterative Optimization



Experiments
- Performance Comparison
- Model Analysis



Conclusion
- Algorithm
- Evaluation

Proposed Algorithm: FEDORA

❑ Federated Parameter Propagation (FEDORA)

$w_{kk'}$: Distribution similarity

$$D_{kk} = \sum_{k'} w_{kk'}$$

- Overall objective function

$$\min_{\theta_k, \hat{\theta}_k} \underbrace{\sum_{k=1}^{K} \frac{1}{\lambda_k n_k} \sum_{i=1}^{n_k} \ell(x_i^k, y_i^k; \theta_k)}_{①} + \underbrace{\sum_{k=1}^{K} \|\theta_k - \hat{\theta}_k\|_2^2}_{②} + \underbrace{\frac{\alpha}{2} \sum_{k=1}^{K} \sum_{k'=1}^{K} \frac{w_{kk'}}{D_{kk}} \left\|\hat{\theta}_k - \hat{\theta}_{k'}\right\|_2^2}_{③}$$

① **Local training**: Each client updates its local parameters θ_k w.r.t. private data

② **Approximation regularization**: Each client approximates the received auxiliary parameters $\hat{\theta}_k$

③ **Distributional regularization:** Two clients share similar auxiliary parameters, if they are distributionally similar

Proposed Algorithm: FEDORA

❑ Federated Parameter Propagation (FEDORA)

- Overall objective function

$$\min_{\theta_k, \hat{\theta}_k} \sum_{k=1}^{K} \frac{1}{\lambda_k n_k} \sum_{i=1}^{n_k} \ell(x_i^k, y_i^k; \theta_k) + \sum_{k=1}^{K} \|\theta_k - \hat{\theta}_k\|_2^2 + \frac{\alpha}{2} \sum_{k=1}^{K} \sum_{k'=1}^{K} \frac{w_{kk'}}{D_{kk}} \left\| \hat{\theta}_k - \hat{\theta}_{k'} \right\|_2^2$$

- Iteratively update the parameters θ_k and $\hat{\theta}_k$

Client update: $$\min_{\theta_k} \frac{1}{n_k} \sum_{i=1}^{n_k} \ell(x_i^k, y_i^k; \theta_k) + \lambda_k \|\theta_k - \hat{\theta}_k\|_2^2$$ (Fix $\hat{\theta}_k$, update θ_k)

Server update: $$\min_{\hat{\theta}_k} \sum_{k=1}^{K} \|\theta_k - \hat{\theta}_k\|_2^2 + \frac{\alpha}{2} \sum_{k=1}^{K} \sum_{k'=1}^{K} \frac{w_{kk'}}{D_{kk}} \left\| \hat{\theta}_k - \hat{\theta}_{k'} \right\|_2^2$$ (Fix θ_k, update $\hat{\theta}_k$)

Training Procedures

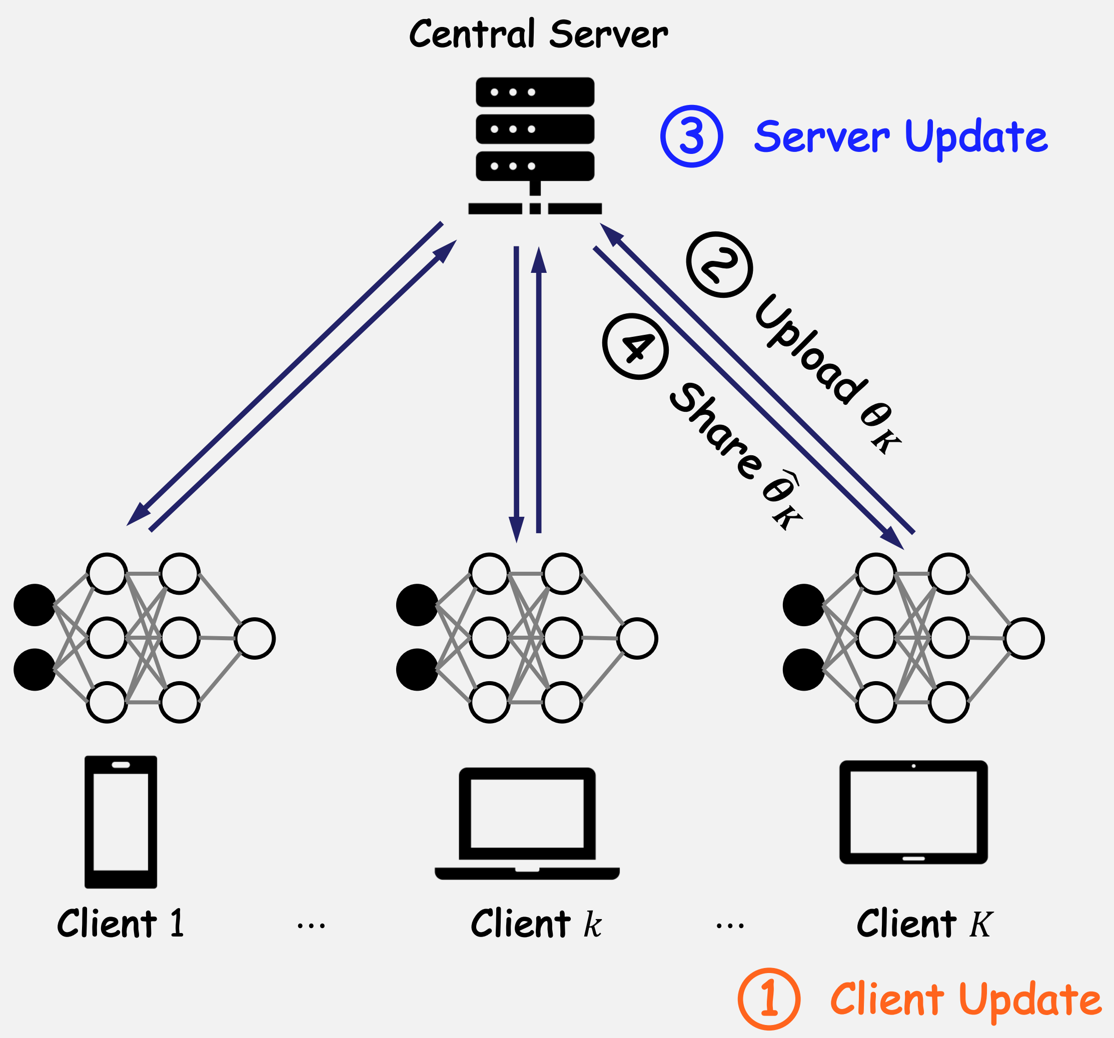
- **Step 1: Client Update**
  - Locally update parameters w.r.t. private data

$$\min_{\theta_k} \frac{1}{n_k}\sum_{i=1}^{n_k} \ell(x_i^k, y_i^k; \theta_k) + \lambda_k \|\theta_k - \hat{\theta}_k\|_2^2$$

- **Step 2: Forward Communication**
  - Upload parameter updates θ_k to the server
- **Step 3: Server Update**:
  - Adaptively aggregate the received parameters

$$\min_{\hat{\theta}_k} \sum_{k=1}^{K} \|\theta_k - \hat{\theta}_k\|_2^2 + \frac{\alpha}{2}\sum_{k=1}^{K}\sum_{k'=1}^{K} \frac{w_{kk'}}{D_{kk}} \left\|\hat{\theta}_k - \hat{\theta}_{k'}\right\|_2^2$$

- **Step 4: Backward Communication**
  - Sent the auxiliary parameters $\hat{\theta}_k$ back to client k



Step 0 – Preprocessing

Distribution Similarity Estimator

- Orthogonal subspace $\mathcal{U}_k$ for client k

(i) **Truncated SVD:** $X_k = U_k \Sigma_k V_k^T$ (Client Update)

- Principal angles between two subspaces

(ii) **Principal Angles:**

$$\zeta_1^{kk'} = \min_{a_1 \in \mathcal{U}_k,\, b_1 \in \mathcal{U}_{k'}} \arccos\left(\frac{\langle a_1, b_1\rangle}{\|a_1\| \cdot \|b_1\|}\right)$$

$$\vdots$$

$$\zeta_p^{kk'} = \min_{\substack{a_1 \in \mathcal{U}_k,\, b_1 \in \mathcal{U}_{k'} \\ a_p \perp a_1, \cdots, a_{p-1} \\ b_p \perp b_1, \cdots, b_{p-1}}} \arccos\left(\frac{\langle a_1, b_1\rangle}{\|a_1\| \cdot \|b_1\|}\right)$$

(Server Update)

- Distribution similarity between client k and client k'

(iii) **Similarity:** $w_{kk'} = \sum_{i=1}^{p} \cos \zeta_i^{kk'}$ (Server Update)

- Saeed Vahidian, et al. "**Rethinking data heterogeneity in federated learning: Introducing a new notion and standard benchmarks**." 2022.

Step 1 – Client Update

Objective Function

- $\hat{\theta}_k$: Encode the knowledge from the central server

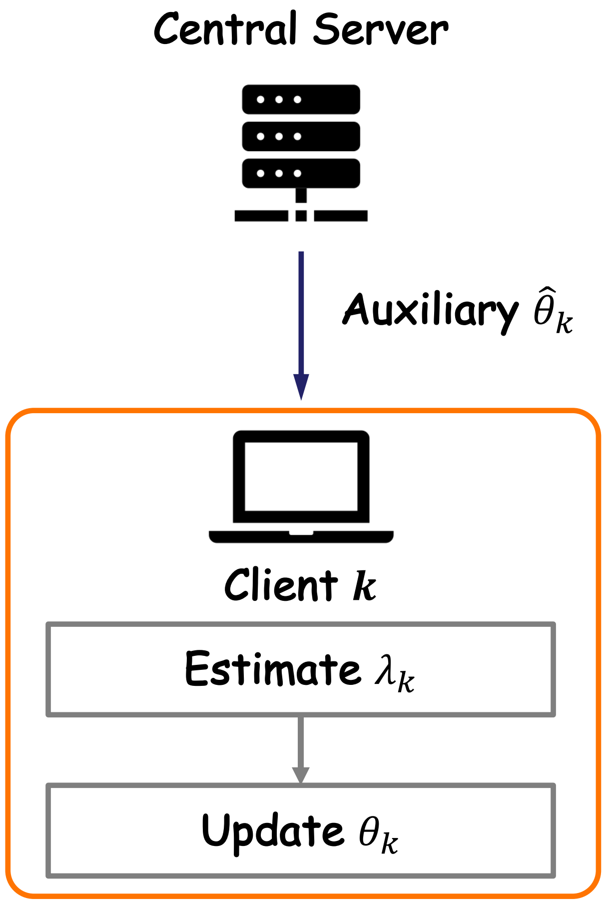
$$\min_{\theta_k} \frac{1}{n_k} \sum_{i=1}^{n_k} \ell(x_i^k, y_i^k; \theta_k) + \lambda_k \|\theta_k - \hat{\theta}_k\|_2^2$$

Selective Regularization

- $\lambda_k = 0 \rightarrow$ pure local training
  $\rightarrow$ a proper λ_k mitigates negative transfer

$$\lambda_k = \max\left(\epsilon, \boxed{\ell_k(\theta_k; D_k^{val}) - \ell_k(\hat{\theta}_k; D_k^{val})}\right) \quad \text{where} \quad \epsilon = 1e-8$$

Source knowledge $\hat{\theta}_k$ enables a smaller generalization error than the target learner θ_k



Step 3 – Server Update

Objective Function

- θ_k: Uploaded personalized parameters from client k

$$\min_{\hat{\theta}_k} \sum_{k=1}^{K} \left\|\theta_k - \hat{\theta}_k\right\|_2^2 + \frac{\alpha}{2}\sum_{k=1}^{K}\sum_{k'=1}^{K} \frac{w_{kk'}}{D_{kk}} \left\|\hat{\theta}_k - \hat{\theta}_{k'}\right\|_2^2$$

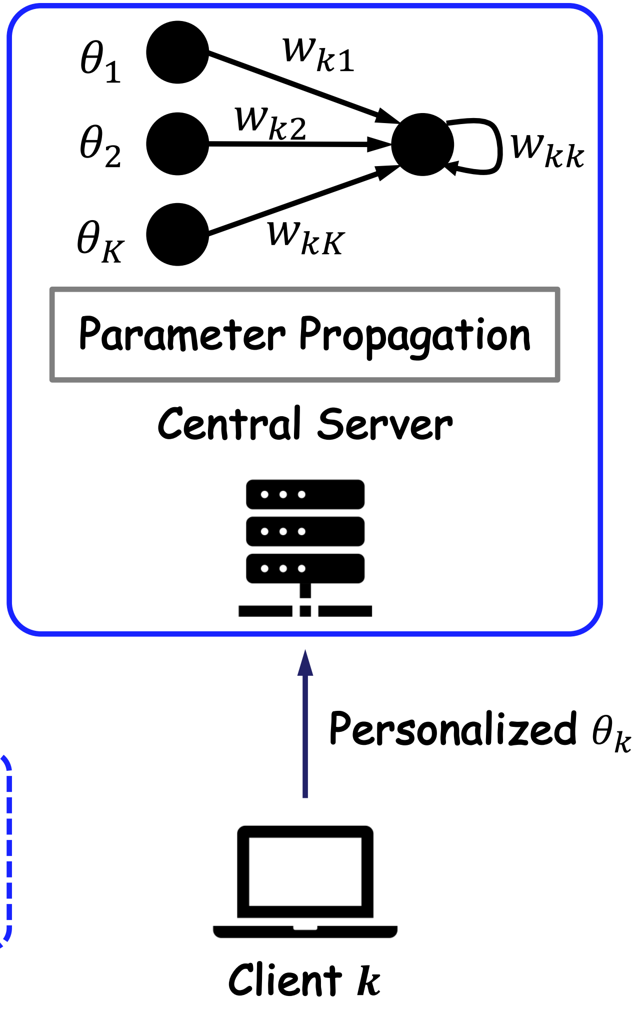
Adaptive Parameter Propagation

- Intuition: Two clients share similar auxiliary parameters, if they are distributionally similar
- An iterative solution:

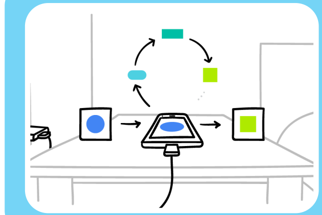
$$\boxed{\hat{\theta}_k^{(m)} = \frac{\alpha}{(1+\alpha)D_{kk}} \sum_{k'=1}^{K} w_{kk'}\hat{\theta}_{k'}^{(m-1)} + \frac{1}{1+\alpha}\theta_k}$$

- A closed-form solution:

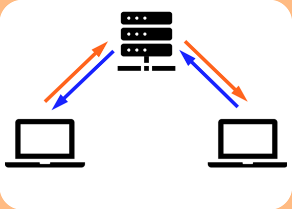
$$\widehat{\Theta}^* = \left(1 - \frac{\alpha}{1+\alpha}\right)\left(I - \frac{\alpha}{1+\alpha}D^{-1}W\right)^{-1}\Theta \quad \text{where} \quad \widehat{\Theta} = [\hat{\theta}_1, \cdots, \hat{\theta}_K]^T$$



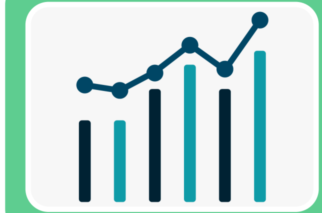
Roadmap



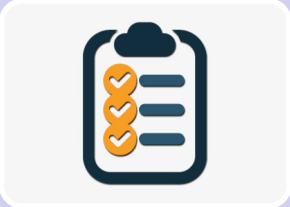
Background
- Personalized Federated Learning
- A Transfer Learning Perspective



Methodology
- Federated Parameter Propagation
- Iterative Optimization



Experiments
- Performance Comparison
- Model Analysis



Conclusion
- Algorithm
- Evaluation

Experiments

❑ Data Sets

- Feature shift: MNIST/Fashion-MNIST/GTSRB
- Label shift: CIFAR10
- Generalized shift: Yearbook

❑ Baselines

- Global FL: FedAvg, FedProx, FedAvg+FT, FedProx+FT
- Local training: LOCAL
- Parameter decoupling: LG-FedAvg, FedPer, pFedHN
- Model interpolation: APFL, Ditto
- Clustering: IFCA, FeSEM
- Multi-task learning: FedFOMO, FedAMP, FedU

❑ Evaluation Metric

- Accuracy

$$\mathrm{ACC}(\theta_k^*) = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} [y_i = y_i^{pred}]$$

- Relative Accuracy

$$\mathrm{R{-}ACC}(\theta_k^*) = \frac{\mathrm{ACC}(\theta_k^*) - \mathrm{ACC}(\theta_k^{LOCAL})}{\mathrm{ACC}(\theta_k^{LOCAL})}$$

- Positive Transferability Ratio

$$\mathrm{PTR} = \frac{1}{K} \sum_{k=1}^{K} \mathbb{I}[\mathrm{ACC}(\theta_k^*) - \mathrm{ACC}(\theta_k^{LOCAL})]$$

- Tian Li, et al. "**Ditto: Fair and robust federated learning through personalization**." ICML 2021.
- Aviv Shamsian, et al. "**Personalized federated learning using hypernetworks**." ICML 2021.
- Michael Zhang, et al. "**Personalized federated learning with first order model optimization**." ICLR 2021.

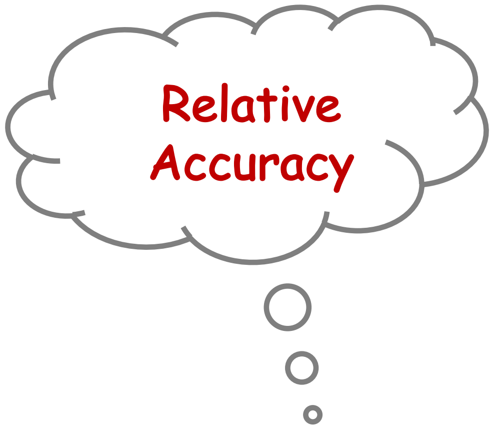
Results under Balanced Setting

- **Balanced Setting**
  - Clients have same number of training samples

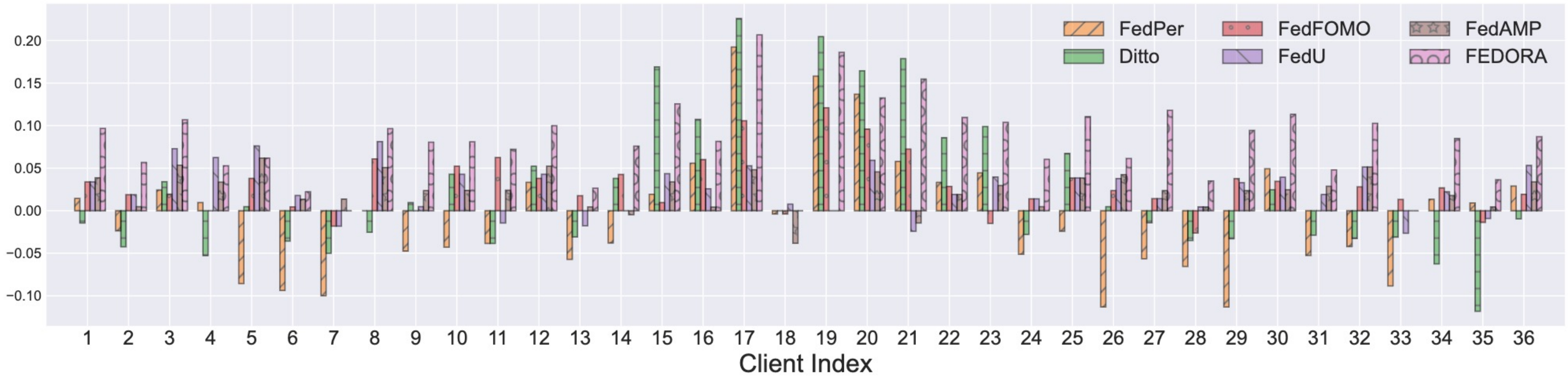
- **Observations:**
  - FEDORA achieves comparable accuracy
  - FEDORA consistently mitigates negative transfer

Model	Rotated MNIST			Rotated Fashion-MNIST			CIFAR-10		
	Acc ↑	R-Acc ↑	PTR ↑	Acc ↑	R-Acc ↑	PTR ↑	Acc ↑	R-Acc ↑	PTR ↑
LOCAL	0.7642	-	-	0.7057	-	-	0.7617	-	-
FedAvg [25]	0.6889	-0.0976	0	0.6441	-0.0847	0.1250	0.6531	-0.1382	0.3000
FedAvg+FT	0.7411	-0.0293	0.3056	0.6848	-0.0283	0.3472	0.7992	0.0513	0.9000
FedProx [21]	0.5375	-0.2962	0	0.5968	-0.1521	0	0.6984	-0.0799	0.2000
FedProx+FT	0.6893	-0.0973	0.0278	0.6788	-0.0358	0.3056	0.7953	0.0460	0.9000
LG-FedAvg [23]	0.7804	0.0214	0.9444	0.7137	0.0115	0.7361	0.7656	0.0054	0.8000
FedPer [1]	0.7741	0.0135	0.6389	0.6725	-0.0457	0.1389	0.8352	0.0990	**1.0000**
pFedHN [33]	0.8004	0.0486	0.8611	0.7215	0.0249	0.6944	0.7766	0.0221	0.6000
APFL [6]	0.7871	0.0303	0.8889	0.7134	0.0112	0.7639	0.8258	0.0866	0.9000
Ditto [20]	0.7806	0.0220	0.7222	0.7212	0.0232	0.7361	0.8078	0.0630	0.9000
IFCA [9]	0.7915	0.0365	0.6944	0.7305	0.0370	0.7639	0.8227	0.0828	0.9000
FeSEM [46]	0.7720	0.0110	0.6111	0.7074	0.0051	0.5278	0.8547	0.1255	**1.0000**
FedFOMO [47]	0.7749	0.0140	0.9167	0.7110	0.0076	0.7639	0.8242	0.0797	**1.0000**
FedU [7]	0.7837	0.0260	0.8889	0.7208	0.0225	0.8056	0.7836	0.0295	0.9000
FedAMP [13]	0.7869	0.0298	**1.0000**	0.7203	0.0213	0.8056	0.7953	0.0457	0.8000
FEDORA	**0.8251**	**0.0806**	**1.0000**	**0.7433**	**0.0548**	**0.9028**	**0.8570**	**0.1288**	**1.0000**

Results under Imbalanced Setting

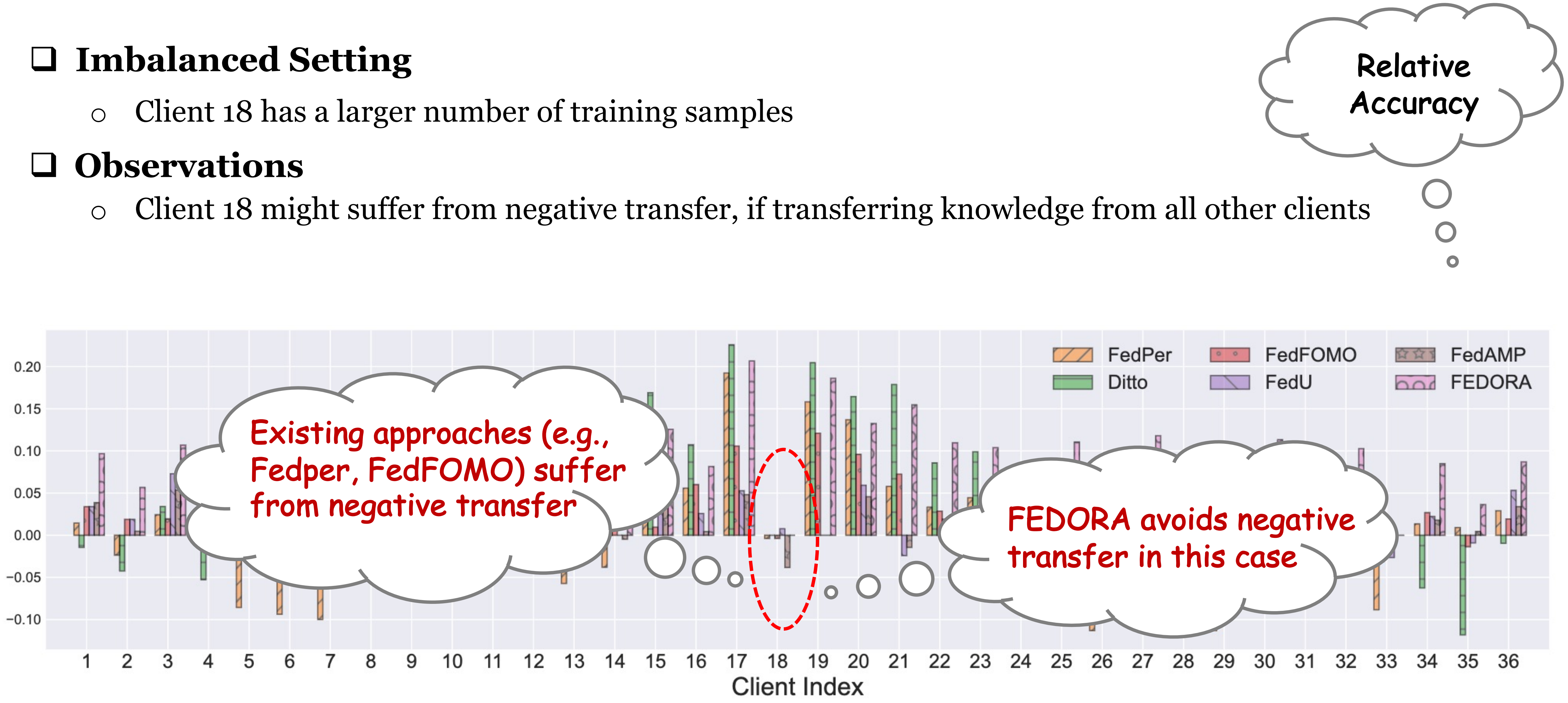


- **Imbalanced Setting**
  - Client 18 has a larger number of training samples



Results under Imbalanced Setting

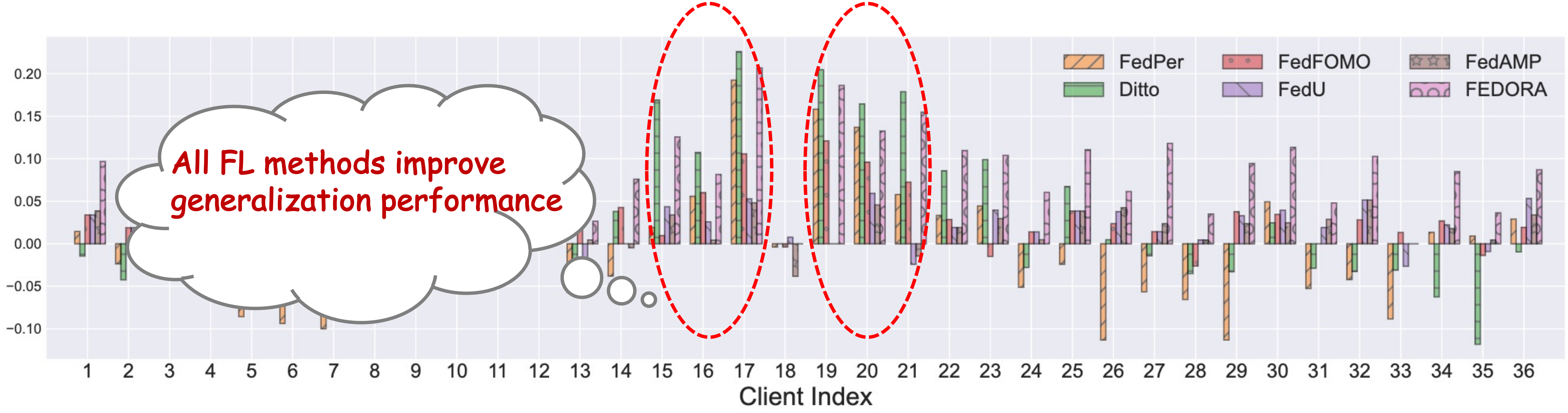
- **Imbalanced Setting**
  - Client 18 has a larger number of training samples
- **Observations**
  - Client 18 might suffer from negative transfer, if transferring knowledge from all other clients



Results under Imbalanced Setting

- **Imbalanced Setting**
  - Client 18 has a larger number of training samples
- **Observations**
  - Client 18 might suffer from negative transfer, if transferring knowledge from all other clients
  - When clients have similar distribution with client 18, they benefit from federated training

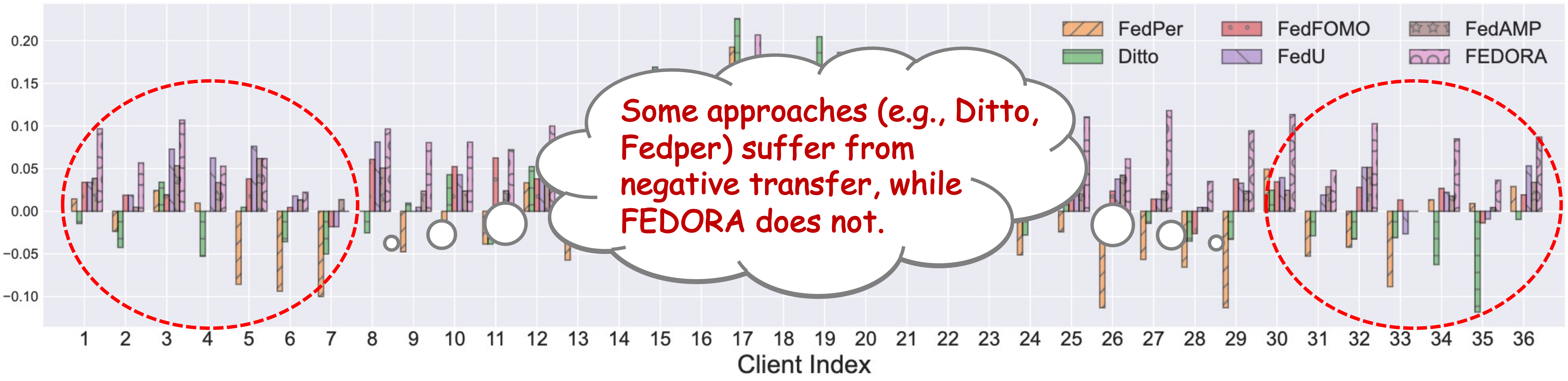
Relative Accuracy



Results under Imbalanced Setting

Relative Accuracy

- **Imbalanced Setting**
  - Client 18 has a larger number of training samples
- **Observations**
  - Client 18 might suffer from negative transfer, if transferring knowledge from all other clients
  - When clients have similar distribution with client 18, they benefit from federated training
  - When clients have different distributions with client 18, they might suffer from negative transfer



Model Analysis

Communication Costs

- FEDORA is comparable with FedAvg

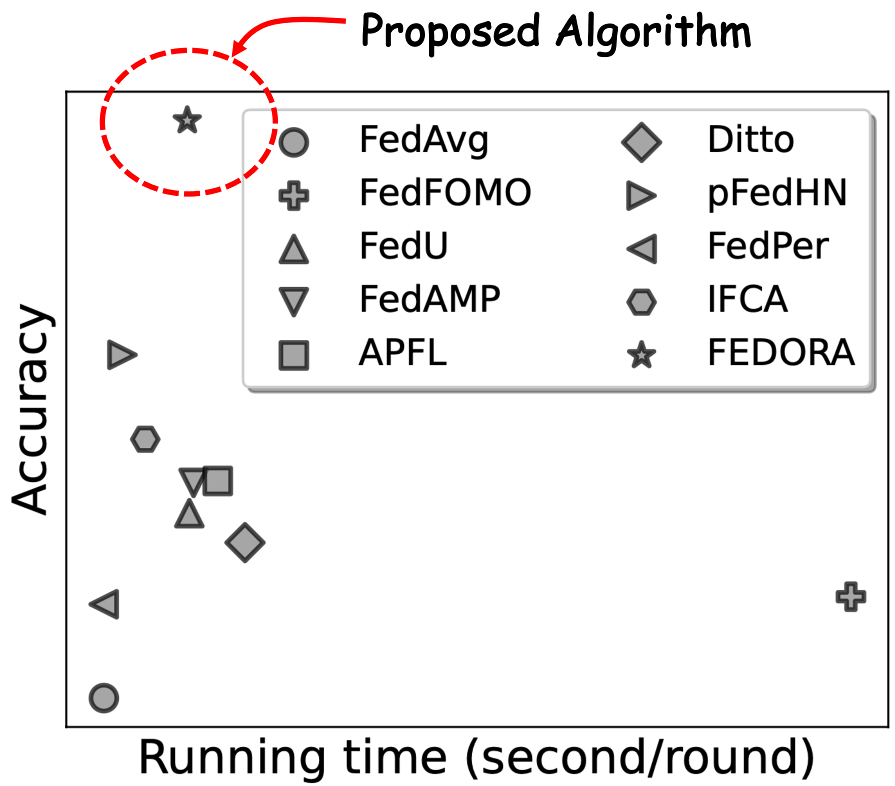
Model	Cost	# params
FedAvg	$2KRd_\theta$	118,282,000
FEDORA	$2KRd_\theta + Kpd_{in}$	118,282,784

Communication costs on Rotated MNIST

K: Number of clients
R: Number of federated training rounds
d_θ: Dimensionality of model parameters
p: Number of orthogonal vectors in the subspace
d_{in}: Dimensionality of the input sample

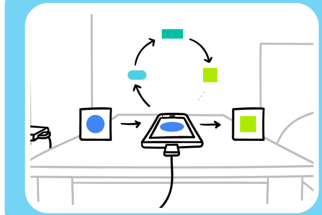
Computational Efficiency

- FEDORA is efficient than other relation-aware pFL algorithms (FedFOMO, FedU, FedAMP)

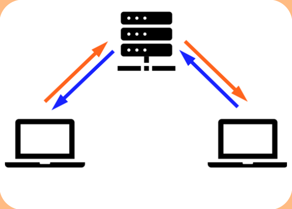


- Brendan McMahan, et al . "**Communication-efficient learning of deep networks from decentralized data**." AISTATS 2017.

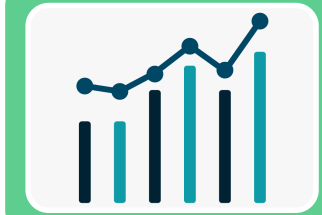
Roadmap



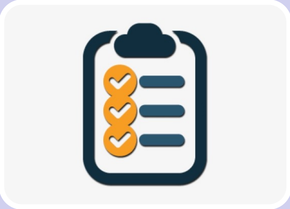
Background
- Personalized Federated Learning
- A Transfer Learning Perspective



Methodology
- Federated Parameter Propagation
- Iterative Optimization



Experiments
- Performance Comparison
- Model Analysis

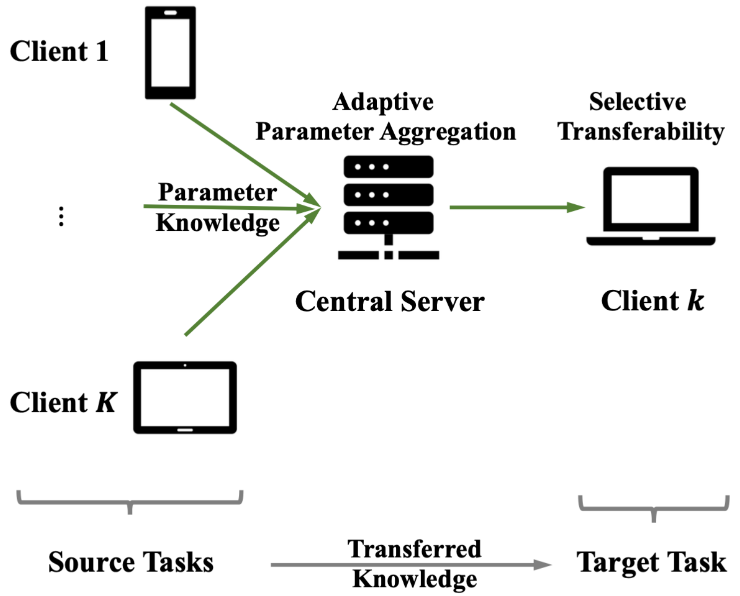


Conclusion
- Algorithm
- Evaluation

Conclusion

❑ Motivation: A Transfer Learning Perspective

- Personalized federated learning suffers from negative transfer



❑ Algorithm: Federated Parameter Propagation

- Adaptive parameter propagation (server update)
- Selective regularization (client update)

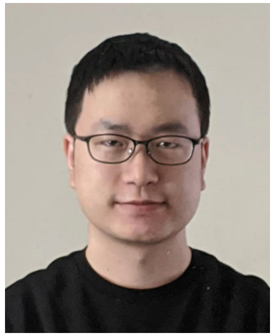
❑ Evaluations

- Effectiveness: Better mitigate the negative transfer
- Efficiency: More efficient than relation-aware pFL baselines
- Communication: Comparable communication cost as FedAvg

Model	Accuracy				Average Accuracy
	Client 1	Client 2	Client 3	Client 4	
LOCAL	0.5270	0.4840	0.4980	0.8110	0.5800
FedAvg	0.3755	0.4420	0.6455	0.7965	0.5649
LG-FedAvg	0.5440	0.5115	0.5430	0.8095	0.6020
Ditto	0.4095	0.4810	0.6465	0.8095	0.5866
FedAMP	0.5300	0.5210	0.5415	0.8105	0.6008
FEDORA	0.5565	0.5675	0.5850	0.8195	0.6321

AIFARMS
Artificial Intelligence for Future Agricultural Resilience, Management, and Sustainability

Personalized Federated Learning with Parameter Propagation




Jun Wu[1] — junwu3@illinois.edu

Wenxuan Bao[1] — wbao3@illinois.edu

Elizabeth Ainsworth[1,2] — ainsworth@illinois.edu

Jingrui He[1] — jingrui@illinois.edu

[1]University of Illinois at Urbana-Champaign
[2]USDA ARS Global Change and Photosynthesis Research Unit

ILLINOIS

USDA Agricultural Research Service