7th AI City Challenge

JUNE 18-22, 2023
CVPR
VANCOUVER, CANADA

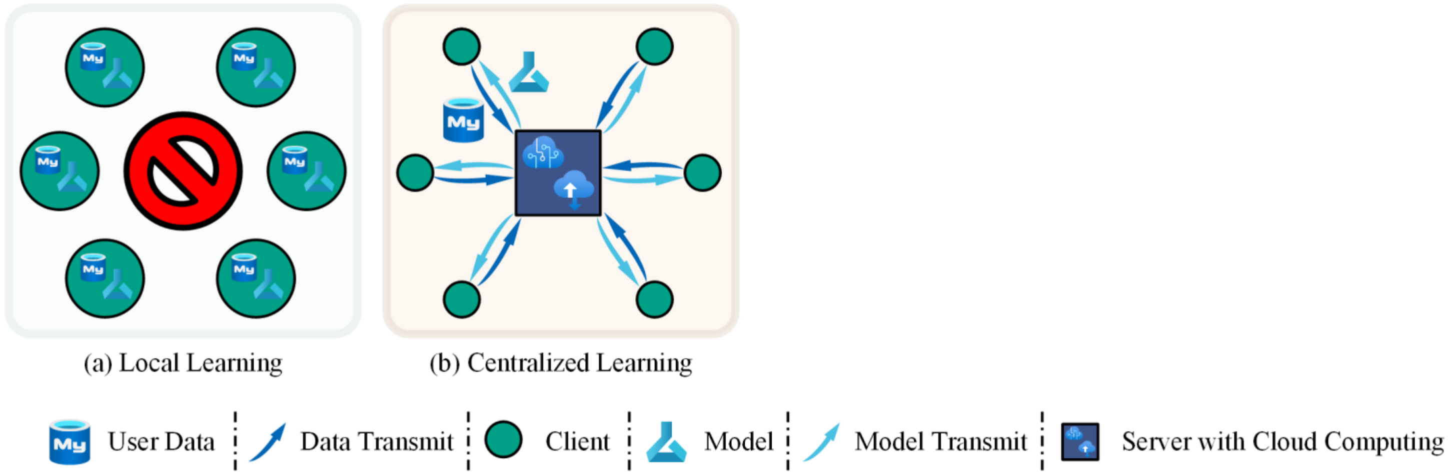
Peer-to-Peer Federated Continual Learning for Naturalistic Driving Action Recognition

Liangqi Yuan, Yunsheng Ma, Lu Su, Ziran Wang

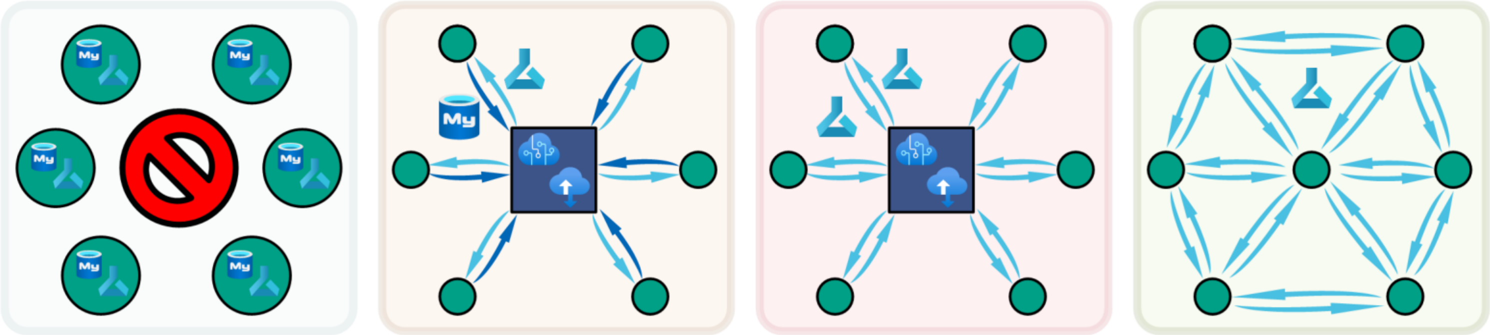
Purdue University

PURDUE UNIVERSITY
Elmore Family School of Electrical and Computer Engineering

Introduction



Introduction



(a) Local Learning (b) Centralized Learning (c) Client-to-Server FL (d) Peer-to-Peer FL

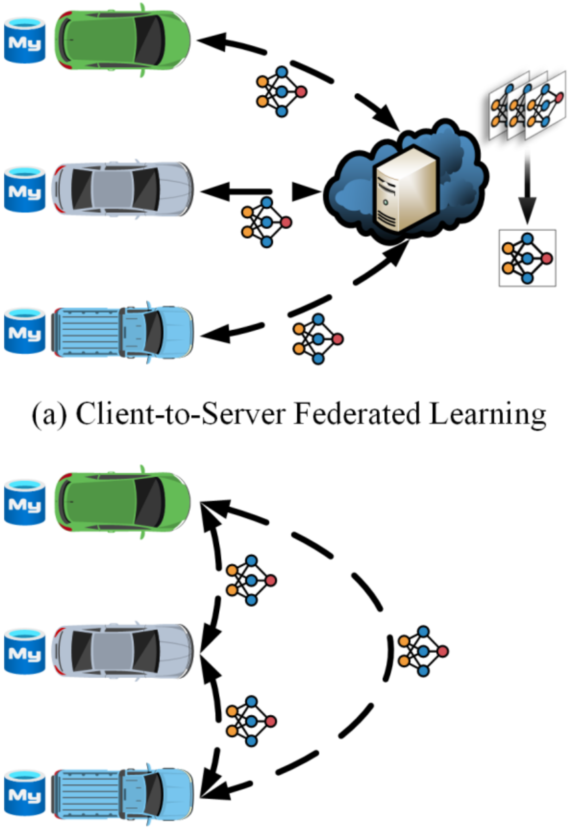
User Data | Data Transmit | Client | Model | Model Transmit | Server with Cloud Computing

Motivation

Concerns about the server:

- Privacy breach
- Fairness
- Security
- Trust
- Single-point of failure

…


(a) Client-to-Server Federated Learning

(b) Peer-to-Peer Federated Learning

Elmore Family School of Electrical and Computer Engineering

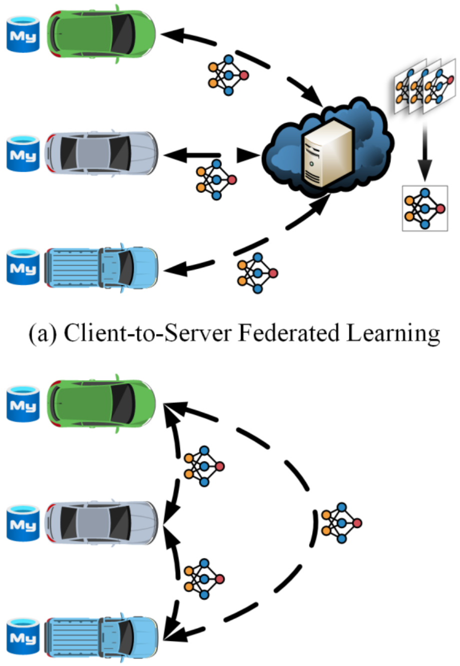
Method

Algorithm 1 FedPC

Input: Iteration rounds (T), client set (C), data set (X_c) and label set (Y_c) for each client ($c \in C$), local training epoch (E), initial model (ω_0), loss function ($\mathcal{L}$), learning rate (η_t)

Output: Trained local models for each client ($\{\omega_c | c \in C\}$)

for $t = 1$ **to** $T - 1$ **do**
  for $c \in C$ **in gossip do**
    Receive the model parameters sent by the previous client $\omega_c \leftarrow \omega_{c-1}$.
    for $e = 1$ **to** $E - 1$ **do**
      Backpropagate the loss function and update the local model $\omega_c^{e+1} \leftarrow \arg\min_{\omega_c^e} \mathcal{L}(\omega_c^e)$.
    end for
    Update the local model $\omega_c \leftarrow \omega_c^E$.
    Client c gossip ω_c to the next client $c + 1$.
  end for
end for



(a) Client-to-Server Federated Learning

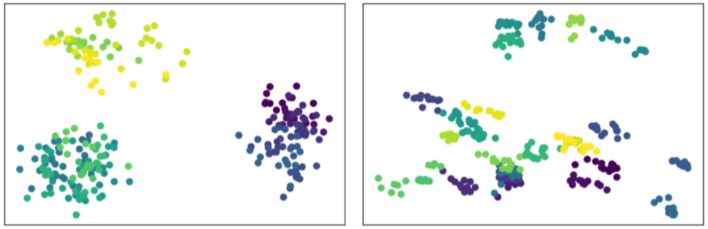
(b) Peer-to-Peer Federated Learning

Objective: $$\min_{\omega} \mathcal{L}(\omega; X_c, Y_c),$$

Optimization: $$\omega_c = \arg\min_{\omega} \mathcal{L}(\omega; X_c, Y_c, \omega_{c-1}).$$

Implementation

Figure 3. Two NDAR datasets are used in the experiments, including (a) StateFarm [33] and (b) AICity [22, 26] datasets.



(a) StateFarm (b) AICity

Figure 4. Data distributions of (a) StateFarm and (b) AICity. The colors denote different clients, and the scatter points represent different activities. The data distributions are visualized by averaging the driver activity images and reducing the dimensions through principal component analysis (PCA).

Implementation

Figure 3. Two NDAR datasets are used in the experiments, including (a) StateFarm [33] and (b) AICity [22, 26] datasets.

Learning Framework: independent learning, FedAvg, FedProx, FedPC (ours)

Base Model: ResNet34

Input: Driver Image

Output: Classification Label

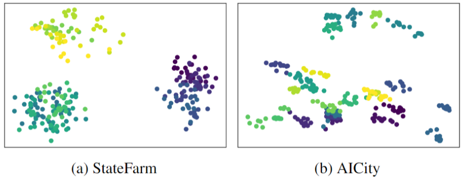


Figure 4. Data distributions of (a) StateFarm and (b) AICity. The colors denote different clients, and the scatter points represent different activities. The data distributions are visualized by averaging the driver activity images and reducing the dimensions through principal component analysis (PCA).

Evaluation

Metric (i) Client Objective. Performance of the current client model on the **local** dataset.

Metric (ii) Generalizability. Performance of the current client model on **other clients'** datasets.

Metric (iii) New Client. Performance of the current client model on **new** clients.

Evaluation

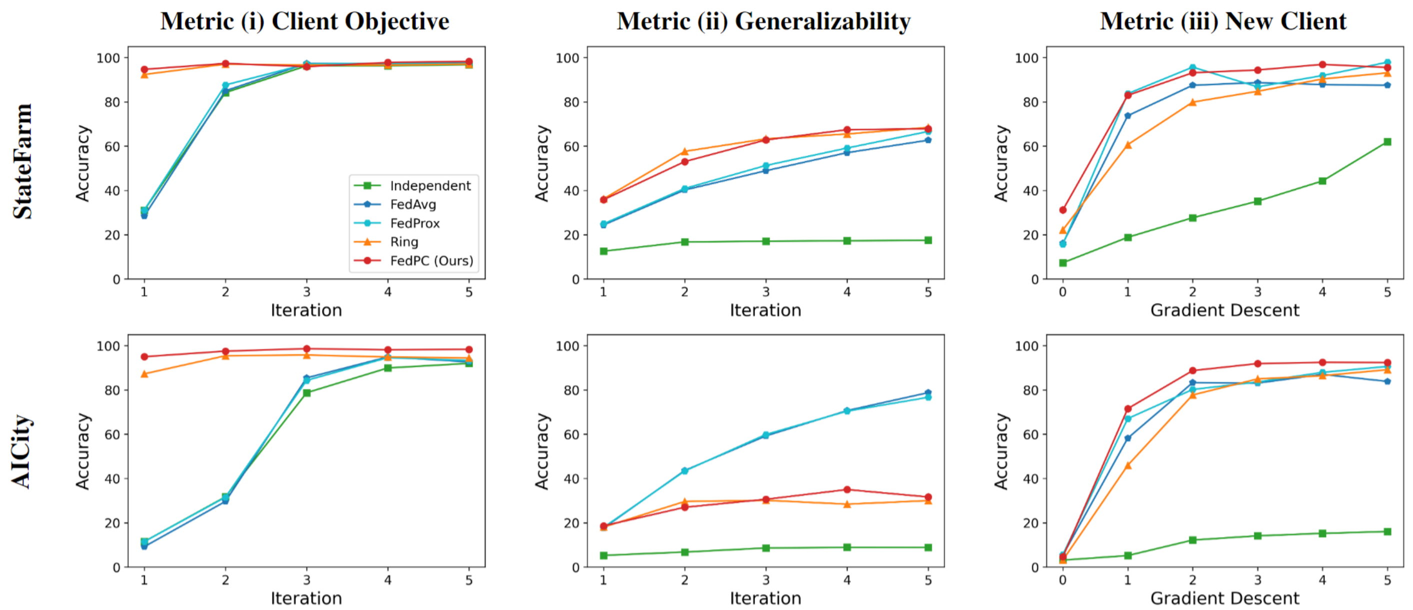


Figure 6. Experimental results on two data sets and three evaluation metrics. The evaluations are performed on the unseen test data set. The data points represent the average of the clients' results.

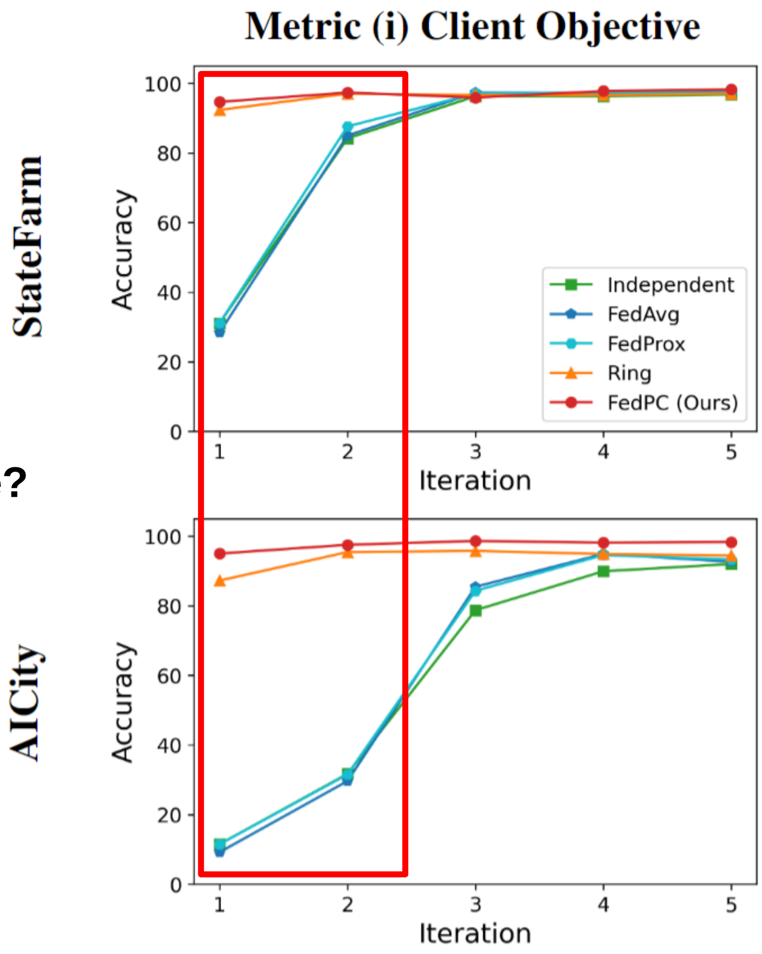
Metric (i) Client Objective. Performance of the current client model on the **local** dataset.

Metric (ii) Generalizability. Performance of the current client model on **other clients'** datasets.

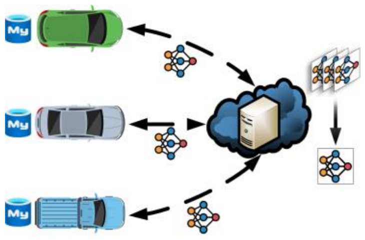
Metric (iii) New Client. Performance of the current client model on **new** clients.

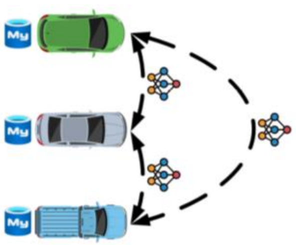
Evaluation

Why is the proposed FedPC effective?



Comparison of FedPC and C2S FL





System	C2S FL (FedAvg)	FedPC (proposed)
Objective	Clients: a personalized model for each client. Server: a single generalized model	Clients: a personalized model for each client. Server: N/A
Knowledge Dissemination	Server aggregation and transmission	Continual learning from another client model
Communication Complexity	Client: send 1 model per iteration round Server: send $\lvert C \rvert$ models per iteration round	Client: send 1 model per iteration round Server: N/A
Dissemination Rate	Slow, it needs to wait for the server to receive, aggregate, and transmit the models	Quick, it only requires clients to transmit the model to each other
Generalizability	Stronger in IID datasets	Partial generalization with non-IID datasets
Compatibility with New Clients	Poor, can be enhanced by personalization	Poor, personalization process may be faster
Hardware Overhead	High, it requires server communication, computing and storage resources	Low
Hidden Concern	Privacy breach, security, trust, SPoF, and aggregation fairness on the server	Lack of incentives, security, and deadlocks on the clients

THANK YOU

Liangqi Yuan, PhD Student
School of Electrical and Computer Engineering
Purdue University
liangqiy@purdue.edu | *www.liangqiy.com*