Federated learning with Flower

김진수

Index

- Federated Learning 소개
- Step by Step Federated Learning
- Method: FedAvg
- FedAvg with flower
- Method: FedProx
- FedProx with flower
- Federated Learning 연구 분야

Federated Learning

- "Communication-Efficient Learning of Deep Networks from Decentralized Data (McMahan et al., google, 2016)
- However, this rich data is often privacy sensitive, large in quantity, or both, which may preclude logging to the data center and training there using conventional approaches.

Federated Learning

- 빠른 속도로 성장하고 있는 연구 분야

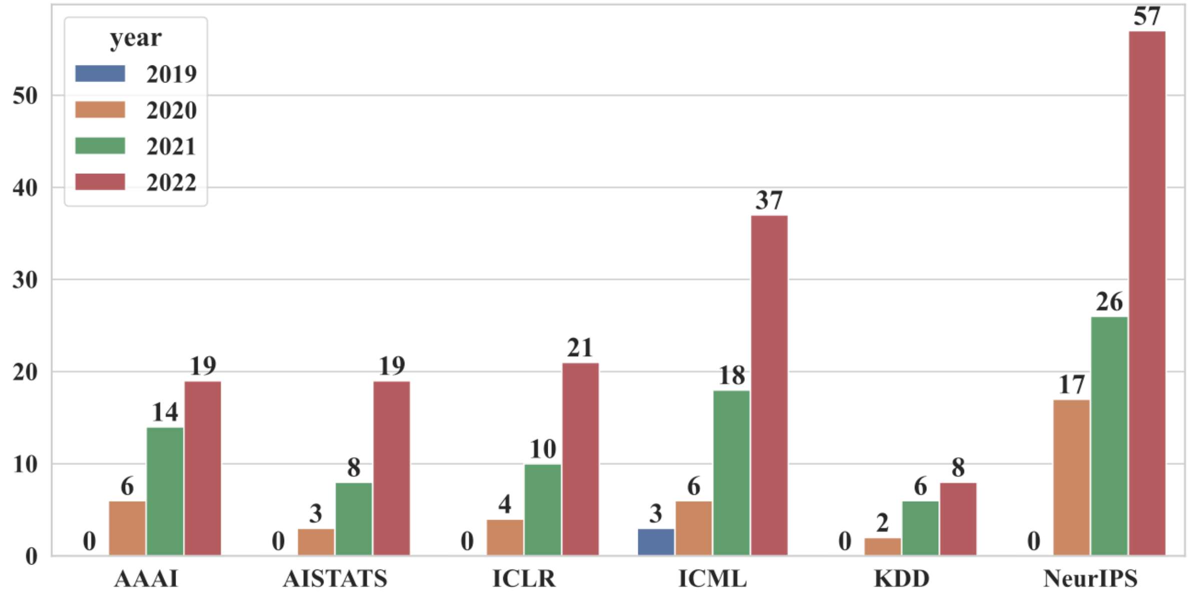
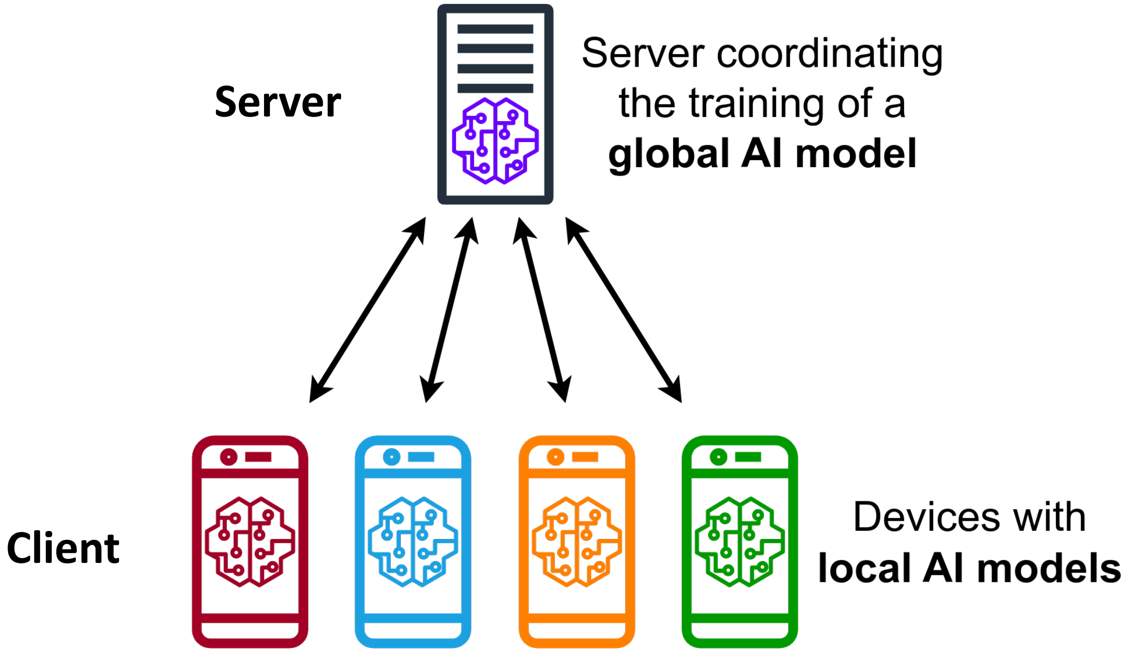


Figure 2: The number of pulished FL papers in top-tie conference from 2019-2022.

"Recent Advances on Federated Learning: A Systematic Survey"(Bingyan Liu, Beijing Univ, 2023)

Step by Step Federated Learning

- Federated Learning은 Server – Client 구조
- Server = 학습되는 모델 구조를 선정하고 로컬 모델을 집계
- Client = Server에게 전달받은 모델을 로컬 데이터를 활용해 학습



Step by Step Federated Learning

1. Server는 학습 모델을 선정(ex. MobileNet, ResNet, AlexNet). $\therefore\ \min_{\mathcal{W}\in\mathbb{R}^d} f(\mathcal{W})$

2. Server는 Client에게 선정한 모델의 초기 Weight를 배포 $\therefore\ Send\ w_0\ to\ Client$

3. Client는 Server로부터 받은 모델을 로컬 데이터로 학습

$\therefore\ C1:\ w_1^1 \leftarrow w_0 - \eta\nabla\mathcal{L}_1(x_1, y_1),\quad C2:\ w_1^2 \leftarrow w_0 - \eta\nabla\mathcal{L}_2(x_2, y_2),\quad C3:\ w_1^3 \leftarrow w_0 - \eta\nabla\mathcal{L}_3(x_3, y_3)$

4. Client는 학습된 로컬 모델의 Weight를 서버로 전송 $\therefore\ return\ w_1^1, w_1^2, w_1^3\ to\ Server$

5. Server는 Client로부터 받은 데이터를 통해 글로벌 모델 업데이트(Aggregation), 라운드 종료

$$\therefore\ w_1 \leftarrow \sum_{k=1}^{3} \frac{n_k}{n} w_1^k$$

Method: FedAvg

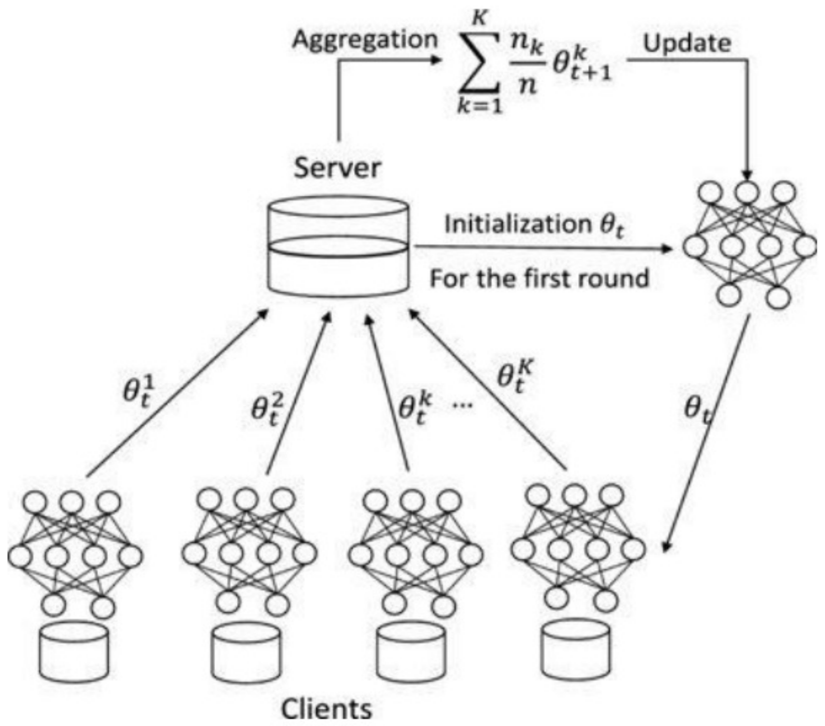
Algorithm 1 `FederatedAveraging`. The K clients are indexed by k; B is the local minibatch size, E is the number of local epochs, and η is the learning rate.

Server executes:
  initialize w_0
  for each round $t = 1, 2, \dots$ **do**
    $m \leftarrow \max(C \cdot K, 1)$
    $S_t \leftarrow$ (random set of m clients)
    for each client $k \in S_t$ **in parallel do**
      $w_{t+1}^k \leftarrow \text{ClientUpdate}(k, w_t)$
    $m_t \leftarrow \sum_{k \in S_t} n_k$
    $w_{t+1} \leftarrow \sum_{k \in S_t} \frac{n_k}{m_t} w_{t+1}^k$ // *Erratum*[4]

ClientUpdate(k, w)**:** // *Run on client* k
  $\mathcal{B} \leftarrow$ (split $\mathcal{P}_k$ into batches of size B)
  for each local epoch i from 1 to E **do**
    for batch $b \in \mathcal{B}$ **do**
      $w \leftarrow w - \eta \nabla \ell(w; b)$
  return w to server

Federated Learning Optimization Problem

$$\min_{\mathcal{W} \in \mathbb{R}^d} f(\mathcal{W}) = \sum_{k=1}^{N} \frac{n_k}{n} f_k(\mathcal{W}, x_k, y_k)$$



FedAvg with Flower

conda create –n gachon_fl python=3.8

git clone --depth=1 https://github.com/adap/flower.git

conda activate gachon_fl

cd flower/baselines

pip install –r requirements.txt

pip install flwr[simulation] 설치가 안된다면 pip install 'ray[tune]’

pip install omegaconf

pip install hydra-core

Method: FedProx

FEDERATED OPTIMIZATION IN $\boxed{\text{HETEROGENEOUS NETWORKS}}$

Tian Li [1] **Anit Kumar Sahu** [2] **Manzil Zaheer** [3] **Maziar Sanjabi** [4] **Ameet Talwalkar** [1 5] **Virginia Smith** [1]

1. Systems Heterogeneity (systems characteristic on each device in network)
2. Statistical Heterogeneity (Non-IID Data)

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

Method: FedProx

Algorithm 1 Federated Averaging (`FedAvg`)

Input: $K, T, \eta, E, w^0, N, p_k, k = 1, \cdots, N$
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 device $k \in S_t$ updates w^t for E epochs of SGD on F_k with step-size η to obtain w_k^{t+1}
  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

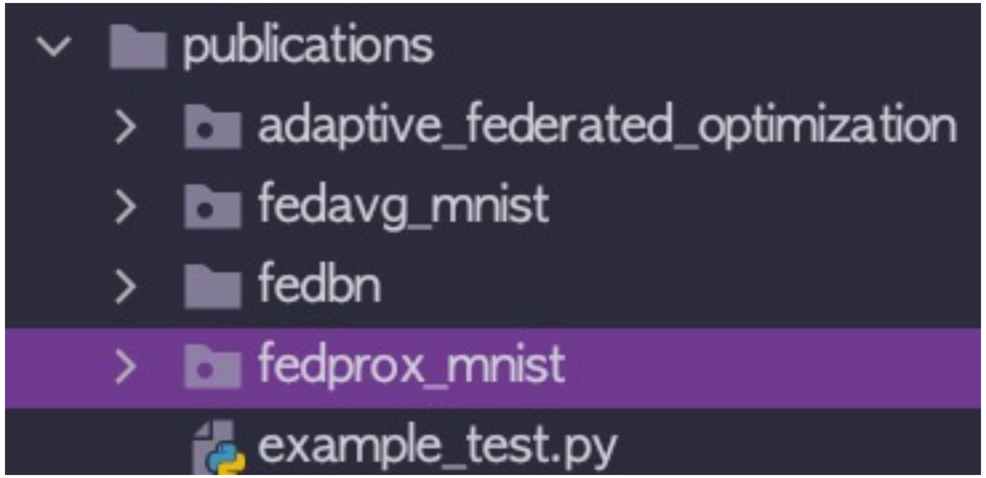
Algorithm 2 `FedProx` (Proposed Framework)

Input: $K, T, \boxed{\mu,} \boxed{\gamma,} w^0, N, p_k, k = 1, \cdots, N$
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 γ_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

Method: FedProx

Definition 2 (γ_k^t-inexact solution)**.** For a function $\boxed{h_k(w; w_t) = F_k(w) + \frac{\mu}{2}\|w - w_t\|^2}$, and $\boxed{\gamma \in [0,1]}$, we say w^* is a γ_k^t-inexact solution of $\min_w h_k(w; w_t)$ if $\boxed{\|\nabla h_k(w^*; w_t)\| \leq \gamma_k^t \|\nabla h_k(w_t; w_t)\|}$, where $\nabla h_k(w; w_t) = \nabla F_k(w) + \mu(w - w_t)$. Note that a smaller γ_k^t corresponds to higher accuracy.

FedProx with flower



Federated Learning 연구분야

Federated Learning Optimization Problem

$$\min_{\mathcal{W}\in\mathbb{R}^d} f(\mathcal{W}) = \boxed{\sum_{k=1}^{N} p_k f_k(\mathcal{W}, x_k, y_k)}$$

Client Selection/Incentive Mechanism

Aggregation optimization

Local Update

$$f_k(\mathcal{W}): \boxed{\mathcal{W}_{t+1}^{k} \leftarrow \mathcal{W}_t - \eta \nabla \mathcal{L}_k(x_k, y_k)}$$

Personalization

Federated Learning 연구분야

- Papers (Research directions)
  - Model Aggregation
  - Personalization
  - Recommender system
  - Security
  - Survey
  - Efficiency
  - Optimization
  - Fairness
  - Application
  - Boosting
  - Incentive mechanism
  - Unsupervised Learning
  - Heterogeneity
  - Client Selection
  - Graph Neural Networks
  - Other Machine Learning Paradigm
  - Trade-off

https://github.com/innovation-cat/Awesome-Federated-Machine-Learning