Federated Learning: System Design and Performance Analysis

Chuan Ma

Nanjing University of Science and Technology

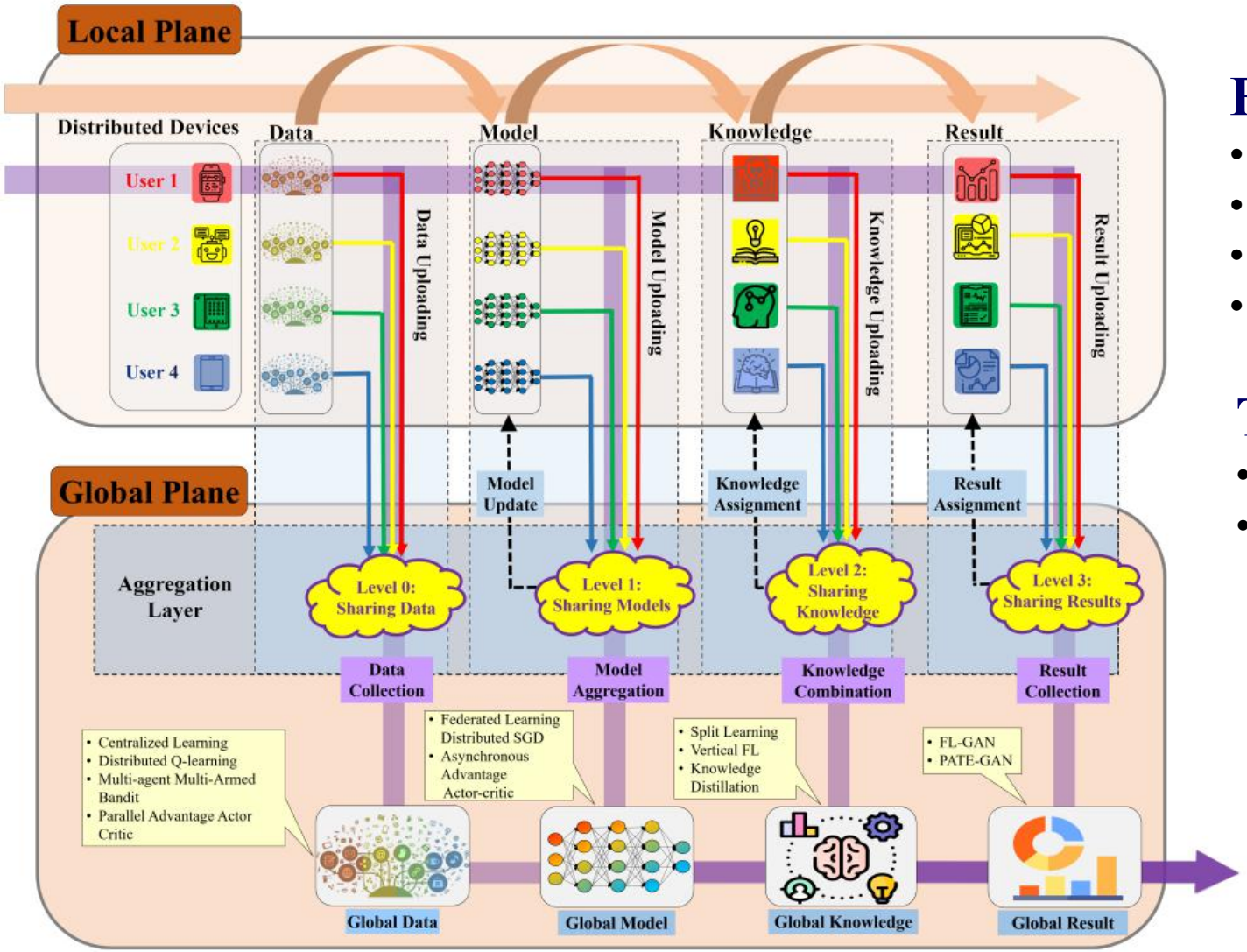
chuan.ma@njust.edu.cn

27th October, 2021

Outline

- Privacy in FL
- Security in FL
- Decentralized FL
- Wireless Communication in FL
- Future Works

Distributed Learning

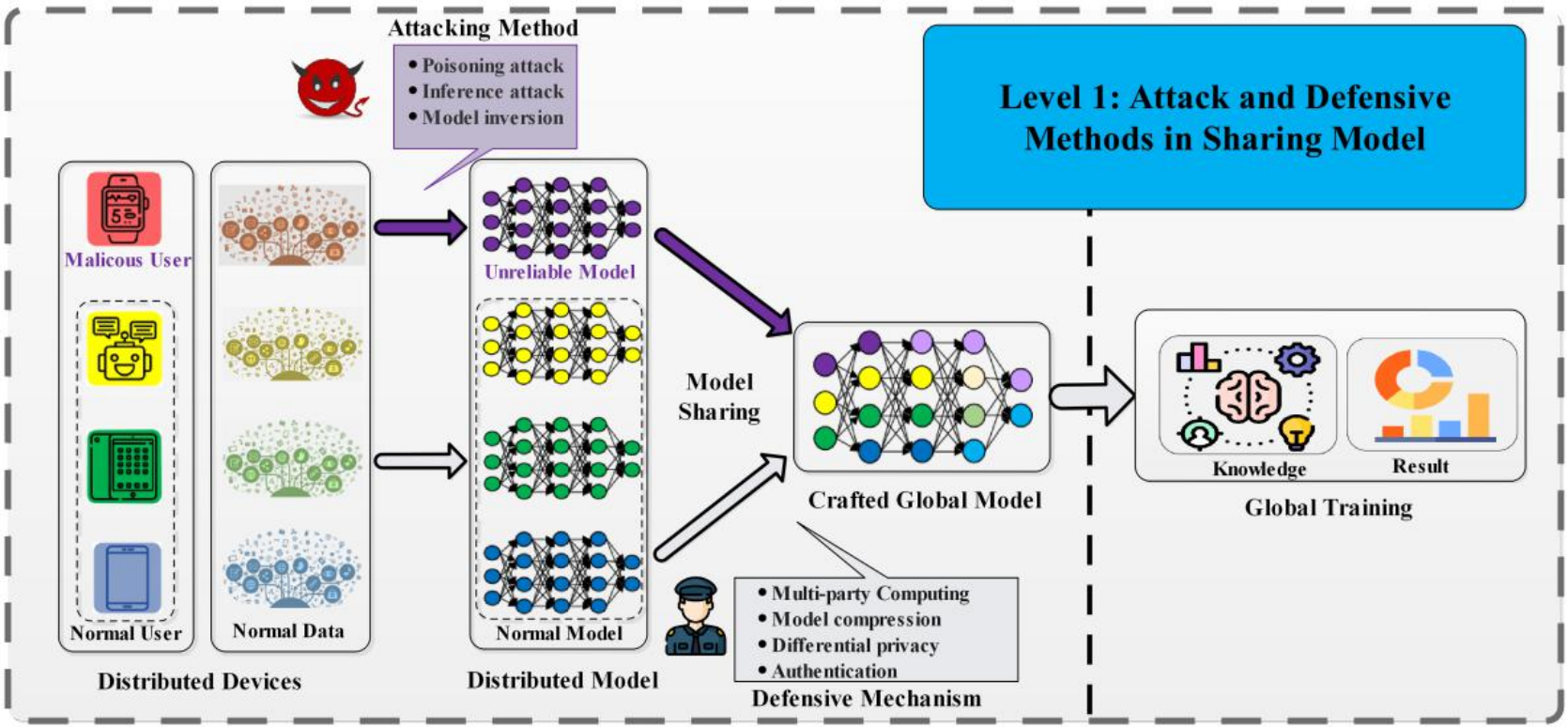


Four Levels:

- Sharing Data
- Sharing Models
- Sharing Knowledge
- Sharing Results

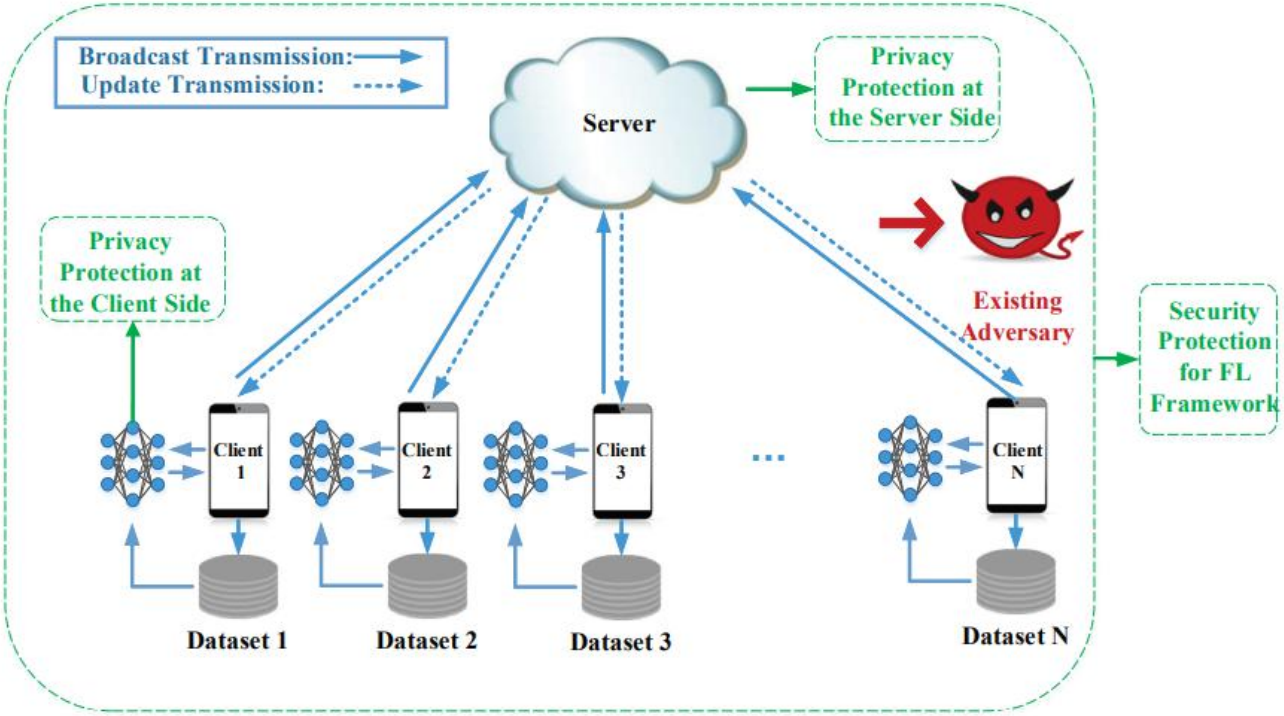
Two Planes:

- Local Plane
- Global Plane



Level 1: Sharing models. All distributed learners need to share their training models with the central server or other participants.

Privacy and Security in FL



Protecting methods

- **Client side privacy**
  - Perturbation
  - Dummy
- **Server side privacy**
  - Robust aggregation
  - Secure multi-party computing
- **Security**
  - Homomorphic encryption
  - Back-door defender

Issues:

- **Convergence v.s. DP**
- **Data/Model poisoning**
- **Scalling up issues**
- **Model aggregation**

- C. Ma et al., "**On Safeguarding Privacy and Security in the Framework of Federated Learning**," in **IEEE Network**, vol. 34, no. 4, pp. 242-248, July/August 2020

Algorithm 1: Noising before Aggregation FL

Data: T, $\mathbf{w}^{(0)}$, μ, ϵ and δ

Initialization: $t = 1$ and $\mathbf{w}_i^{(0)} = \mathbf{w}^{(0)}, \forall i$

while $t \leq T$ **do**

  Local training process:

  while $\mathcal{C}_i \in \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_N\}$ **do**

    Update the local parameters $\mathbf{w}_i^{(t)}$ as

$$\mathbf{w}_i^{(t)} = \arg\min_{\mathbf{w}_i} \left(F_i(\mathbf{w}_i) + \frac{\mu}{2} \|\mathbf{w}_i - \mathbf{w}^{(t-1)}\|^2 \right)$$

    Clip the local parameters

$$\mathbf{w}_i^{(t)} = \mathbf{w}_i^{(t)} / \max\left(1, \frac{\|\mathbf{w}_i^{(t)}\|}{C}\right)$$

    Add noise and upload parameters

$$\widetilde{\mathbf{w}}_i^{(t)} = \mathbf{w}_i^{(t)} + \mathbf{n}_i^{(t)}$$

  Model aggregating process:

  Update the global parameters $\mathbf{w}^{(t)}$ as

$$\mathbf{w}^{(t)} = \sum_{i=1}^{N} p_i \widetilde{\mathbf{w}}_i^{(t)}$$

  The server broadcasts global noised parameters

$$\widetilde{\mathbf{w}}^{(t)} = \mathbf{w}^{(t)} + \mathbf{n}_{\mathrm{D}}^{(t)}$$

  Local testing process:

  while $\mathcal{C}_i \in \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_N\}$ **do**

    Test the aggregating parameters $\widetilde{\mathbf{w}}^{(t)}$ using local dataset

  $t \leftarrow t + 1$

Result: $\widetilde{\mathbf{w}}^{(T)}$

Key step:

- Adding noise to the parameters **after clipping before aggregation** to achieve DP

Key value:

- C: the clipping threshold, which is related the sensitivity (different with Google)

Lemma 1 (Sensitivity after the aggregation operation): *In FL training process, the sensitivity for $\mathcal{D}_i$ after the aggregation operation $s_{\mathrm{D}}^{\mathcal{D}_i}$ is given by*

$$\Delta s_{\mathrm{D}}^{\mathcal{D}_i} = \frac{2Cp_i}{m}. \tag{8}$$

- **n**: a Gaussian noise follows $N(0, \sigma^2)$

- K. Wei et al., "**Federated Learning With Differential Privacy: Algorithms and Performance Analysis**," in **IEEE Transactions on Information Forensics and Security**, vol. 15, pp. 3454-3469, 2020, doi: 10.1109/TIFS.2020.2988575.

Theorem 2 (Convergence upper bound of the NbAFL): *With required protection level ϵ, the convergence upper bound of* **Algorithm 1** *after T aggregations is given by*

$$\mathbb{E}\{F(\widetilde{\mathbf{w}}^{(T)}) - F(\mathbf{w}^*)\} \leq P^T\Theta + \left(\frac{\kappa_1 T}{\epsilon} + \frac{\kappa_0 T^2}{\epsilon^2}\right)(1 - P^T), \quad (17)$$

where

$$P = 1 + 2l\lambda_2,\ \kappa_1 = \frac{\lambda_1 \beta c}{m(1-P)}\sqrt{\frac{2}{N\pi}} \quad (18)$$

and

$$\kappa_0 = \frac{\lambda_0 c^2}{m^2(1-P)N}. \quad (19)$$

- A large ε ⟶ A small loss
- A large N ⟶ A small loss
- **Optimal *T*** exists in certain N and ε

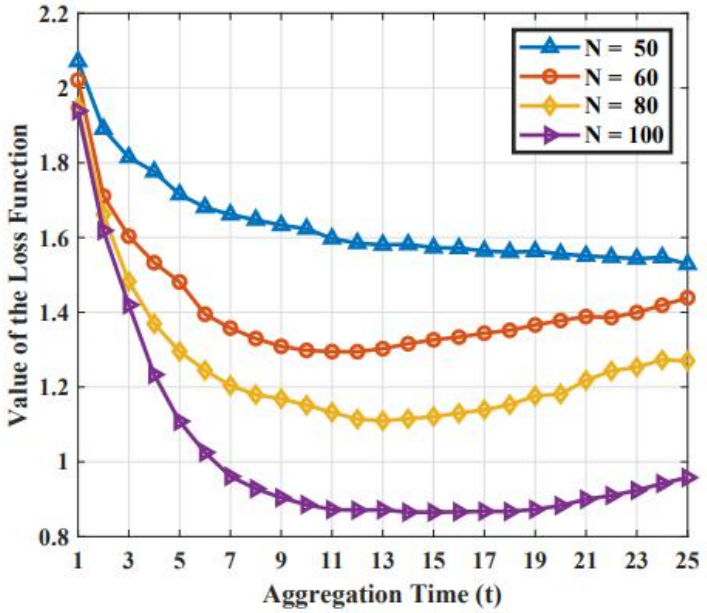


Figure 6: The value of the loss function with various numbers of clients under $\epsilon = 60$ under NbAFL Algorithm with 50 clients.

K-client random scheduling FL

Theorem 3 (Convergence under K-random scheduling): *With required protection level ϵ and the number of chosen clients K, for any $\Theta > 0$, the convergence upper bound after T aggregation times is given by*

$$\mathbb{E}\{F(\widetilde{\mathbf{v}}^T) - F(\mathbf{w}^*)\} \leq Q^T\Theta + \frac{1-Q^T}{1-Q}\left(\frac{c\alpha_1\beta}{-mK\ln\left(1-\frac{N}{K}+\frac{N}{K}e^{-\frac{\epsilon}{T}}\right)}\sqrt{\frac{2}{\pi}} + \frac{c^2\alpha_0}{m^2K^2\ln^2\left(1-\frac{N}{K}+\frac{N}{K}e^{-\frac{\epsilon}{T}}\right)}\right). \quad (23)$$

where

$$Q = 1 + \frac{2l}{\mu^2}\left(\frac{\rho B^2}{2} + \rho B + \frac{\rho B^2}{K} + \frac{2\rho B^2}{\sqrt{K}} + \frac{\mu B}{\sqrt{K}} - \mu\right), \quad (24)$$

$$\alpha_0 = \frac{2\rho K}{N} + \rho,\ \alpha_1 = 1 + \frac{2\rho B}{\mu} + \frac{2\rho B\sqrt{K}}{\mu N}, \quad (25)$$

and

$$\widetilde{\mathbf{v}}^{(T)} = \sum_{i=1}^{K} p_i\left(\mathbf{w}_i^{(T)} + \mathbf{n}_i^{(T)}\right) + \mathbf{n}_{\mathrm{D}}^{(T)}. \quad (26)$$

- In each communication round, K clients are randomly chosen from N clients to participant the learning
- **Optimal K** exists

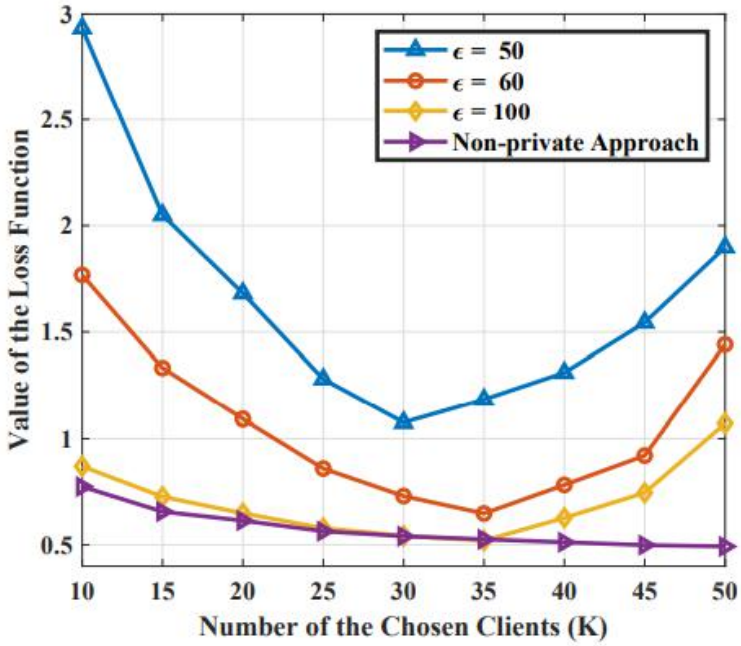


Figure 11: The value of the loss function with various numbers of chosen clients under $\epsilon = 50, 60, 100$ under NbAFL Algorithm and non-private approach with 50 clients.

- K. Wei et al., "**Federated Learning With Differential Privacy: Algorithms and Performance Analysis**," in **IEEE Transactions on Information Forensics and Security**, vol. 15, pp. 3454-3469, 2020, doi: 10.1109/TIFS.2020.2988575.

Motivation:

In practice, how to determine the optimal number *T* of the communication rounds?

Note: it is **NOT** practical to get the optimal *T* from the theoretical expression because it requires running the ML training first to obtain a few parameter values to be used in the derived upper bound.

Thoughts:

- We want to check the ML convergence performance on the fly.
- We want to dynamically reduce the value of *T*, in order to avoid using a too large *T*.
- We want to **dynamically adjust the DP noise variance** according to the dynamic *T* to achieve a certain privacy budget.
- By adjusting the DP noise variance, we want to gradually reduce the noise variance as the ML training converges.

- K. Wei et al., "**User-Level Privacy-Preserving Federated Learning: Analysis and Performance Optimization**," in **IEEE Transactions on Mobile Computing**, doi: 10.1109/TMC.2021.3056991.

Privacy-Preserving Federated Learning: Algorithm

Algorithm 2: UDP with CRD Method

Input: The value of an initial T, LDP parameters (ϵ_i, δ_i), clipping threshold C and discounting factor β ($\beta < 1$).

Initialize: $t = 0$ and $\boldsymbol{w}^0$
while $t < T$ **do**
  Broadcast: $\boldsymbol{w}^t$ and T to all MTs
  for $\forall i \in \mathcal{K}$ **do**
    Calculate the STD of additive noises using (25);
    Locally train with clipping gradients: $\boldsymbol{w}_i^{t+1} = \ell(\mathcal{D}_i, \boldsymbol{w}^t)$;
    Add (ϵ_i, δ_i)-LDP noise: $\widetilde{\boldsymbol{w}}_i^{t+1} = \boldsymbol{w}_i^t + \mathcal{N}(0, \sigma_i^t \mathbf{I})$;
    Upload noised parameters to the server;
  Aggregate received model parameters: $\boldsymbol{w}^{t+1} = \sum_{i \in \mathcal{K}} p_i \widetilde{\boldsymbol{w}}_i^{t+1}$;
  for $\mathcal{C}_i \in \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_U\}$ **do**
    Test the aggregating parameters $\boldsymbol{w}^{t+1}$; using local dataset
  if $\mathcal{V}(\boldsymbol{w}^t) - \mathcal{V}(\boldsymbol{w}^{t+1}) < \zeta$ **then**
    Update the preset value of T: $T = \lfloor \beta(T - t) \rfloor + t$;
  $t = t + 1$;
return $\boldsymbol{w}^T$

Key steps:

- We want to check the ML convergence performance (between consecutive rounds) on the fly.

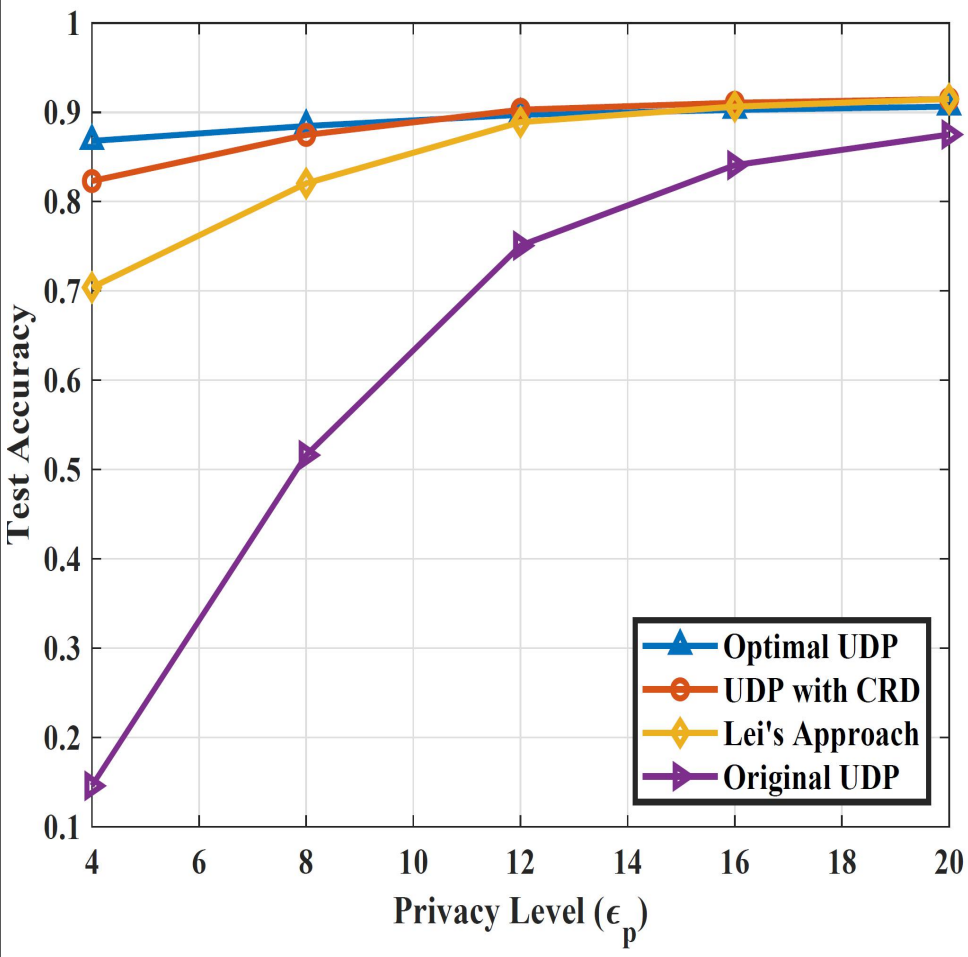
Step 8: *Communication rounds discounting:* When the convergence performance stops improving by the following decision $\mathcal{V}(\boldsymbol{w}^t) - \mathcal{V}(\boldsymbol{w}^{t+1}) < \zeta$ and ζ is the threshold, the discounting method will be triggered in the server, where $\mathcal{V}(\boldsymbol{w}^t)$ is the test loss by the model $\boldsymbol{w}^t$. The server will obtain a smaller T than

- We want to dynamically adjust/recalibrate the DP noise variance according to the dynamic T to achieve a certain privacy budget.

Theorem 4. After t ($0 \leq t < T$) communication rounds and a new T, the STD of additive noises for the i-th MT to guarantee (ϵ_i, δ_i)-LDP can be given as

$$\sigma_i^t = \left(\frac{T - t}{\frac{\epsilon_i^2}{2q\Delta\ell^2 \ln\left(\frac{1}{\delta_i}\right)} - \sum_{\tau=0}^{t-1} \frac{1}{(\sigma_i^\tau)^2}} \right)^{\frac{1}{2}}. \quad (25)$$

Privacy-Preserving Federated Learning: Results



- **Original UDP** (user-level DP): Google's work in 2016 [A. Martin, et al., "Deep learning with differential privacy," CCS'16]. Equal noise across T=200 communication rounds.
- **Lei's approach** [L. Yu, et al., "Differentially private model publishing for deep learning," S&P'19]. The STD of added noise will be reduced linearly until the privacy budget is spent over T=200 communication rounds.
- **Optimal UDP** (user-level DP). An optimal T is obtained using the theoretical expression in our previous TIFS paper. Not practical. Need to run the ML training first to obtain a few parameter values to be used in the theoretical expression.

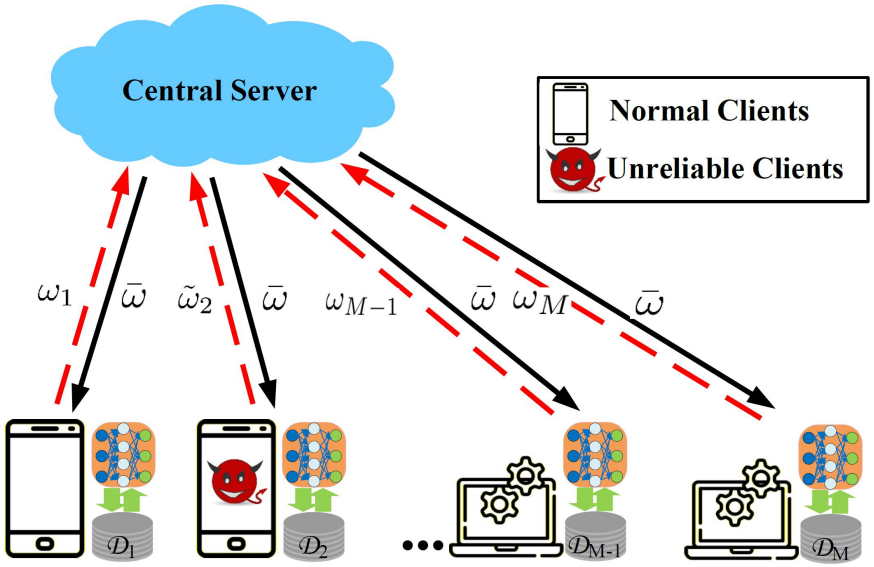
Other works

- Ma C, Li J, Ding M, et al. **RDP-GAN: A Rényi-Differential Privacy based Generative Adversarial Network[J]**. arXiv preprint arXiv:2007.02056, 2020.
  - **DP in GAN**
  - Adding noise on the loss functions of the discriminator to achieve DP, while using Rényi-DP to measure the privacy level
  - Dynamic decreasing the injecting noise
  - Submitting to TDSC, under major revision

- Ma C, Yuan L, et al. **Data Level Privacy Preserving: A Stochastic Perturbation Approach based on Differential Privacy[J]**
  - **Data level privacy preservation**
  - Produce random noise based on other records
  - Submitting to TKDE, under major revision

Motivation:

- we recast the research of private FL in another name of robust/secure FL?
  - ➢ Specifically, noise is not assumed to be added in data processing to protect individuals' privacy
  - ➢ Instead, noise is introduced by imperfect devices/sensors or malicious parties, and we need to address the noise issue in FL



Unreliable/Malicious client:

$$\widehat{\boldsymbol{w}}_i^{(k\tau)} = \alpha \boldsymbol{w}_i^{(k\tau)} + \boldsymbol{n}_i^{(k\tau)}, \tag{4}$$

- **Unreliable behaviors:** noisy local models
  - ➢ For example, set $|\boldsymbol{n}_i| > 0, \alpha = 0.9 \sim 1$
- **Malicious behaviors:** aiming to sabotage the global model aggregation
  - ➢ For example, set $\alpha = -1$ (completely reversing the signs of local models) or $|\boldsymbol{n}_i| > 0$, $0 < \alpha < 0.5$ (useless local models)

Probabilistic Unreliability and Model Aggregation:

$$\bar{\boldsymbol{w}}^{(k\tau)} = \sum_{i \in M} p_i \tilde{\boldsymbol{w}}_i, \tag{5}$$

where

$$\tilde{\boldsymbol{w}}_i = \begin{cases} \boldsymbol{w}_i, & \text{with probability } 1 - P_{\mathrm{U}}; \\ \alpha \boldsymbol{w}_i + \boldsymbol{n}_i, & \text{with probability } P_{\mathrm{U}}, \end{cases} \tag{6}$$

- C. Ma, J. Li, M. Ding, K. Wei, W. Chen and H. V. Poor, "**Federated Learning with Unreliable Clients: Performance Analysis and Mechanism Design**," in **IEEE Internet of Things Journal**, doi: 10.1109/JIOT.2021.3079472.

FL with unreliable/malicious clients: **Convergence Analysis**

Theorem 1 *For some $\varepsilon > 0$ and $\Theta > 0$, when the clients in the FL system behave unreliable with probability p_{U}, the convergence upper bound with a fixed total number of iterations T is given by*

$$\mathbb{E}\left\{F\left(\bar{\boldsymbol{w}}^{(T)}\right)\right\} - F(\boldsymbol{w}^*) \leq \frac{1}{T\left(\omega\eta(1-\frac{\beta\eta}{2}) - \frac{\rho\left(\phi(\tau)+\frac{p_{\mathrm{U}}}{M}\left[(1-\alpha)M\Theta+\frac{2\sqrt{M}\sigma}{\pi}\right]\right)}{\tau\varepsilon^2}\right)}, \quad (7)$$

where $\phi(\tau) = \frac{\delta}{\beta}\left((\eta\beta+1)^{\tau}-1\right) - \eta\delta\tau$, $\varphi = \omega(1-\frac{\beta\eta}{2})$, *and* $\omega \triangleq \min_k \frac{1}{\|\boldsymbol{w}^{(k\tau)}-\boldsymbol{w}^*\|^2}$.

Proof: See Appendix A. ■

The upper bound given by **Theorem 1** demonstrates the convergence result of the FL system with unreliable clients. A lower bound means that the value of the system loss function converges closer to the optimal one.

Notation	Description
$\boldsymbol{w}_i^{(k\tau)}$	The uploaded model of i-th client at the k-th communication round
τ	The number of local training epochs
$F_i(\cdot)$	The local objective function of the i-th client
$\widehat{\boldsymbol{w}}_i^{(k\tau)}$	The local model of the i-th unreliable client
α	The scaling factor with range [-1,1]
$\boldsymbol{n}$	The additive noise
p_{U}	The probability of the unreliable behavior
p_i	The aggregating weight based on the training data size of $\mathcal{D}_i$
T	The number of total training iterations
k	The number of total aggregation

Assumption 1 *We assume the following conditions are satisfied for all i, $\forall i \in \mathcal{M}$:*

1) $F_i(\boldsymbol{w})$ *is convex;*
2) *All model parameters satisfy* $\|\boldsymbol{w}\| \leq \Theta$*;*
3) $F_i(\boldsymbol{w})$ *is ρ-Lipschitz, i.e.,* $\|F_i(\boldsymbol{w}) - F_i(\boldsymbol{w}')\| \leq \rho\|\boldsymbol{w} - \boldsymbol{w}'\|$*, for any $\boldsymbol{w}$, $\boldsymbol{w}'$;*
4) $F_i(\boldsymbol{w})$ *is β-smooth, i.e.,* $\|\nabla F_i(\boldsymbol{w}) - \nabla F_i(\boldsymbol{w}')\| \leq \beta\|\boldsymbol{w} - \boldsymbol{w}'\|$*, for any $\boldsymbol{w}$, $\boldsymbol{w}'$;*
5) $\eta \leq \frac{1}{\beta}$*, where η is the step size;*
6) $\|F(\boldsymbol{w}^t) - F(\boldsymbol{w}^*)\| \geq \varepsilon$*, for all $\boldsymbol{w}$ during FL training.*

- C. Ma, J. Li, M. Ding, K. Wei, W. Chen and H. V. Poor, "**Federated Learning with Unreliable Clients: Performance Analysis and Mechanism Design**," in **IEEE Internet of Things Journal**, doi: 10.1109/JIOT.2021.3079472.

- **Basic idea**: Train a discriminator neural network to detect the unreliable clients and discard them in the model aggregation

- **Intuition 1**: For reliable clients, a certain level of correlation should exist in consecutive model updates
- => Such correlation can be picked up by a **CNN** neural network, which should be effective in mitigating noisy models

- **Intuition 2**: For reliable clients, their models should look much different from those with reversed signs
- => Such difference can be detected by an **MLP** neural network, which should be effective in mitigating malicious models

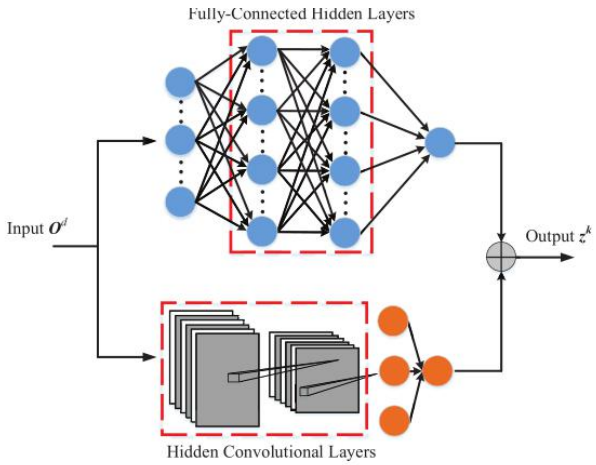


Fig. 2. The architecture of the DNN based detector.

FL with unreliable/malicious clients: **Results**

TABLE II
THE CLASSIFICATION ACCURACY COMPARISON IN MNIST/FASHION MNIST DATASET WITH DIFFERENT UNRELIABLE PROBABILITIES.

	All reliable	Deep-SA	Krum	Secprobe	Pearson
Case I	0.88/0.85	**0.81/0.78**	0.71/0.69	0.78/0.75	0.74/0.71
Case II	0.88/0.85	**0.8/0.765**	0.72/0.69	0.77/0.74	0.76/0.72
Case III	0.88/0.85	**0.81/0.77**	0.73/0.7	0.78/0.75	0.74/0.7

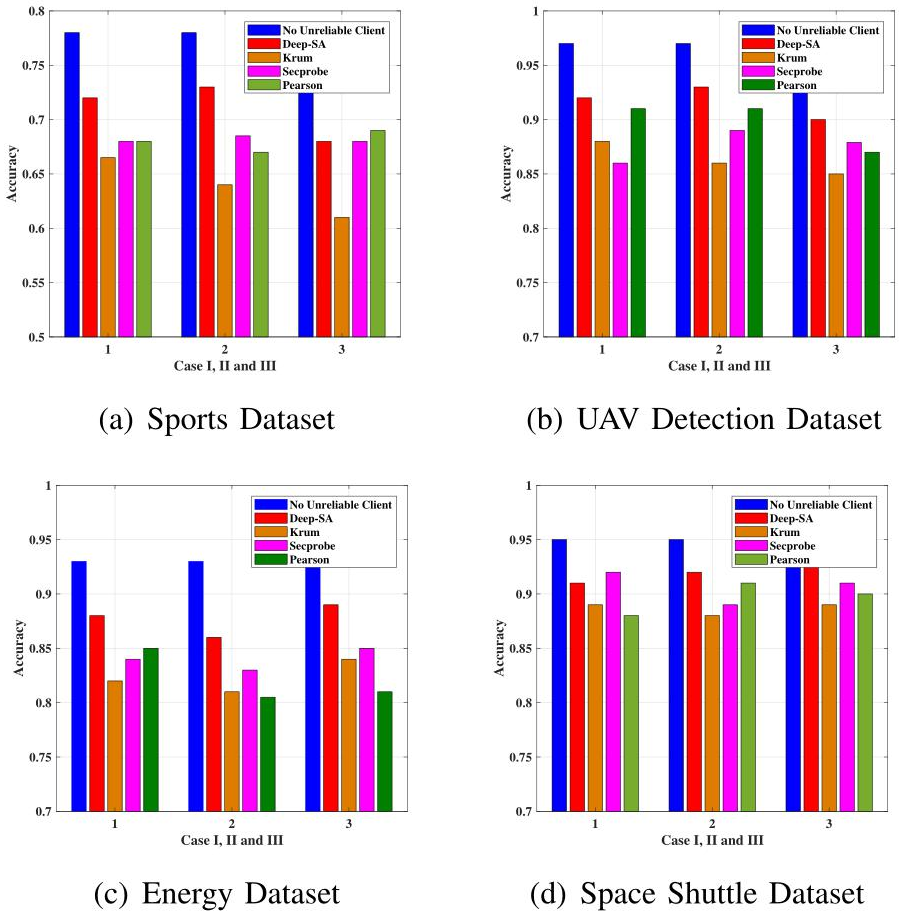
TABLE III
THE CLASSIFICATION ACCURACY COMPARISON IN CIFAR-10/ADULT DATASET WITH DIFFERENT UNRELIABLE PROBABILITIES.

	All reliable	Deep-SA	Krum	Secprobe	Pearson
Case I	0.68/0.89	**0.58/0.82**	0.51/0.77	0.57/0.8	0.54/0.78
Case II	0.68/0.89	**0.59/0.83**	0.52/0.79	0.57/0.81	0.56/0.79
Case III	0.68/0.89	**0.6/0.83**	0.52/0.78	0.58/0.81	0.54/0.78

- The multivariate classification accuracy performance for 3 Cases. The baseline of FL with all reliable clients yields the upper-bound performance (**the 1st col.**).
- The results were collected using the **MNIST, Fashion-MNIST, CIFAR-10, and Adult datasets**.
- The proposed Deep-SA algorithm(**the 2nd col.**) achieves the best performance in all cases compared with the Krum, Secprobe, and Pearson schemes.

- C. Ma, J. Li, M. Ding, K. Wei, W. Chen and H. V. Poor, "**Federated Learning with Unreliable Clients: Performance Analysis and Mechanism Design**," in **IEEE Internet of Things Journal**, doi: 10.1109/JIOT.2021.3079472.

FL with unreliable/malicious clients: **Results**



(a) Sports Dataset (b) UAV Detection Dataset

(c) Energy Dataset (d) Space Shuttle Dataset

Fig. 8. The classification accuracy comparison between the proposed Deep-SA algorithm and others.

- The multivariate classification accuracy performance for 3 cases. The baseline of FL with all reliable clients yields the upper-bound performance (**blue bars**).
- The results were collected using the Sports, UAV, Energy, and Space Shuttle datasets.
- The proposed Deep-SA algorithm (**red bars**) achieves the best performance in most cases compared with the Krum, Secprobe, and Pearson schemes.

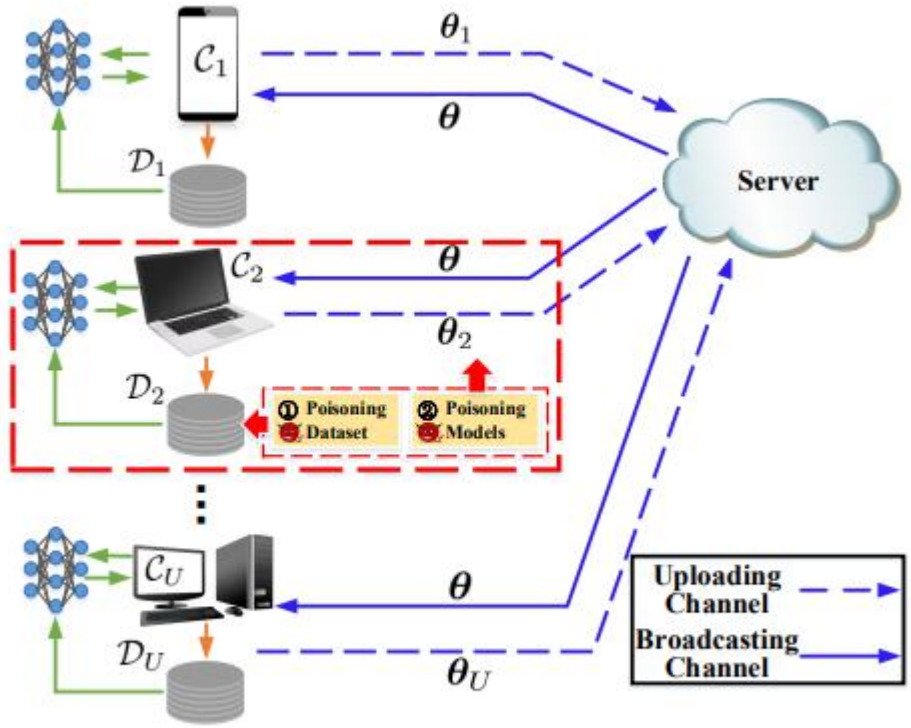


Fig. 1. Data poisoning attacks vs. model poisoning attacks.

Macilious clients

Goal:

- Untargeted: degrade the learning performance
- Targeted: produce designed predictions

Background knoweledge:

- Full: everything
- Partial: aggregation rules, global model and updates model from compromised clients
- No： updates model from compromised clients

Participant Collusion:

- Coordinate the poisoned updates

- Wei K, Li J, Ding M, et al. **Covert Model Poisoning Against Federated Learning: Algorithm Design and Optimization**[J]. arXiv preprint arXiv:2101.11799, 2021, submitted to **TDSC**.

Full and Partial Knowledge attacking:

- **Know the aggregation rule**

Mean Aggregation Rule:

$$\widehat{\boldsymbol{\theta}} = \sum_{i \in \boldsymbol{\mathcal{B}}} p_i \boldsymbol{\theta}_i + \sum_{i' \in \boldsymbol{\mathcal{M}}} p_{i'} \widehat{\boldsymbol{\theta}}_{i'},$$

$M + B = U$. We define $\boldsymbol{\mathcal{M}}$ as the set of these compromised clients, $\boldsymbol{\mathcal{B}}$ as the set of benign clients, and $\boldsymbol{\mathcal{U}}$ as the set of all clients, where $\boldsymbol{\mathcal{M}} \subseteq \boldsymbol{\mathcal{U}}$ and $\boldsymbol{\mathcal{B}} = \boldsymbol{\mathcal{U}} / \boldsymbol{\mathcal{M}}$. In details, in

- **Exist the collusion clients**

Attacking Goal:

$$\begin{aligned} \widehat{\boldsymbol{\theta}}^*_{\boldsymbol{\mathcal{M}}} = & \underset{\widehat{\boldsymbol{\theta}}_{i'} \subseteq \boldsymbol{\Theta}, i' \in \boldsymbol{\mathcal{M}}}{\arg\min} \; F_A\left(\widehat{\boldsymbol{\theta}}\right), \\ \text{s.t.} \quad & \widehat{\boldsymbol{\theta}} = \mathcal{A}(\widehat{\boldsymbol{\theta}}_{i'}; \boldsymbol{\theta}_i), \forall i' \in \boldsymbol{\mathcal{M}}, i \in \boldsymbol{\mathcal{B}}, \end{aligned} \tag{4}$$

where $\mathcal{A}$ represents the aggregation rule and $\widehat{\boldsymbol{\theta}}^*_{\boldsymbol{\mathcal{M}}} \triangleq \{\widehat{\boldsymbol{\theta}}^*_{i'} | i' \in \boldsymbol{\mathcal{M}}\}$ represents the optimal poisoning models.

Note that $\mathbf{F_A}$ denotes the objective functions of the attackers.

The optimal solution for the attacking models:

Theorem 1 *With a certain target $F_A(\boldsymbol{\theta})$ and M compromised clients under the mean aggregation rule in FL, the crafted model can be calculated by*

$$\widehat{\boldsymbol{\theta}}_i = \frac{1}{\sum_{i \in \boldsymbol{\mathcal{M}}} p_i} \left(\widehat{\boldsymbol{\theta}}^* + \left(\frac{2}{\sum_{i \in \boldsymbol{\mathcal{M}}} p_i} - 1 \right) \sum_{i \in \boldsymbol{\mathcal{M}}} p_i \boldsymbol{\theta}_i \right), \forall i \in \boldsymbol{\mathcal{M}}, \tag{6}$$

and the loss function value can be expressed as

$$F_A\left(\widehat{\boldsymbol{\theta}}\right) = \left(\frac{2}{\sum_{i \in \boldsymbol{\mathcal{M}}} p_i} - 1 \right)^2 \left\| \sum_{i \in \boldsymbol{\mathcal{M}}} p_i \boldsymbol{\theta}_i \right\|^2. \tag{7}$$

Proof: See Appendix A. □

Remark: By estimating the local models of the benign clients, we can obtain the optimal attacking models.

Aggregation rule Krum with FKB and PKB

Krum [22] selects one of the U local models that is the most similar to other models as the global model. The

Algorithm 1 Original Covert Model Poisoning

Require: $\mathcal{D}_{\text{att}} = \mathcal{D}_{\mathcal{U}}$ (full knowledge), $\mathcal{D}_{\text{att}} = \mathcal{D}_{\mathcal{M}}$ (partial knowledge), σ, η_0, ε and λ.

$t \leftarrow 0$ (communication round counter)
while $t < T$ **do**
  The compromised clients craft models as follows:
  $\boldsymbol{\theta}^t_{\text{att}} = \boldsymbol{\theta}^t_{\mathcal{U}}$ (if full knowledge) and
  $\boldsymbol{\theta}^t_{\text{att}} = \boldsymbol{\theta}^t_{\mathcal{M}}$ (if partial knowledge)
  $\widehat{\boldsymbol{\theta}}^t_{1,k} \leftarrow \boldsymbol{\theta}^t$ and $k \leftarrow 0$ (iteration counter)
  repeat
    $\widehat{\boldsymbol{\theta}}^t_{1,k+1} \leftarrow \Pi_{\boldsymbol{\Theta}}\left(\widehat{\boldsymbol{\theta}}^t_{1,k} - \eta_k \nabla_{\widehat{\boldsymbol{\theta}}_1} F_A(\widehat{\boldsymbol{\theta}}^t_{1,k})\right)$
    for $i = 2, 3, \ldots, M$ **do**
      $\boldsymbol{n}_i \leftarrow \mathcal{N}(0, \sigma)$
      $\widehat{\boldsymbol{\theta}}^t_{i,k+1} \leftarrow \widehat{\boldsymbol{\theta}}^t_{1,k+1} + \varepsilon \boldsymbol{n}_i / \|\boldsymbol{n}_i\|$
    end for
    $\widehat{\boldsymbol{\theta}}^t_{k+1} \leftarrow \mathcal{A}_{\text{krum}}(\widehat{\boldsymbol{\theta}}^t_{i'}; \boldsymbol{\theta}^t_i), \forall i' \in \boldsymbol{\mathcal{M}}, i \in \boldsymbol{\mathcal{B}}$
    if $\widehat{\boldsymbol{\theta}}^t_{1,k+1} \neq \widehat{\boldsymbol{\theta}}^t_{k+1}$ **then**
      $\eta_{k+1} \leftarrow \lambda \eta_k$
      $\widehat{\boldsymbol{\theta}}^t_{1,k+1} \leftarrow \widehat{\boldsymbol{\theta}}^t_{1,k}$
    end if
    $k \leftarrow k + 1$
  until $\eta_k < \varsigma$
end while

No knowledge background

Algorithm 2 Original Covert Model Poisoning with No Knowledge Background (CMP-NKB)

Require: $\mathcal{D}_{\text{att}} = \mathcal{D}_{\mathcal{M}}$, η^0 and λ.

$t \leftarrow 0$ (communication round counter)
while $t < T$ **do**
  The server broadcasts the aggregated model $\boldsymbol{\theta}^t$
  The compromised clients craft models as follows:
  if $\|\widehat{\boldsymbol{\theta}}^t_1 - \boldsymbol{\theta}^t\| \leq \xi$ **then**
    $\eta^{t+1} \leftarrow \lambda \eta^t$
  else
    $\eta^{t+1} \leftarrow \eta^t / \lambda$
  end if
  $\widehat{\boldsymbol{\theta}}^t_1 \leftarrow \Pi_{\boldsymbol{\Theta}}\left(\boldsymbol{\theta}^t - \eta^t \nabla_{\boldsymbol{\theta}^t} F_A(\boldsymbol{\theta}^t; \mathcal{D}_{\text{att}})\right)$
  for $i = 2, 3, \ldots, M$ **do**
    $\boldsymbol{n}_i \leftarrow \mathcal{N}(0, \sigma)$
    $\widehat{\boldsymbol{\theta}}^t_i \leftarrow \widehat{\boldsymbol{\theta}}^t_i + \varepsilon \boldsymbol{n}_i / \|\boldsymbol{n}_i\|$
  end for
  All clients upload the local models to the server
  The server aggregate uploaded models by
  a certain aggregation rule
  $t \leftarrow t + 1$
end while

Benchmarks:

- **Fang's FKB and PKB** ("Local model poisoning attacks to byzantine-robust federated learning", USENIX Security, 2020)
- **Arjun' attack** ("Analyzing Federated Learning through an Adversarial Lens", ICML, 2019)

TABLE 3
The comparison of test accuracy between our proposed untargeted attack algorithms, i.e. the prop. CMP-PKB, the prop. CMP-FKB and existing algorithms on SVM model and MNIST dataset under various degrees of non-i.i.d. with $U = 50$, $M = 10$ and $T = 30$.

The Degree of Non-i.i.d.	Approach	Error Rate (%)	Successful Attacking Rate (%)	Time Cost (second)
$p = 0$	The Prop. CMP-FKB-Orgcontr	42.498	100	3.357
	The Prop. CMP-PKB-Orgcontr	22.201	92.0	1.268
	Fang's FKB	28.714	100	0.077
	Fang's PKB	16.863	82.6	0.115
	Gaussian Attack	15.510	0.00	0.00073
$p = 0.5$	The Prop. CMP-FKB-Orgcontr	43.254	100	3.338
	The Prop. CMP-PKB-Orgcontr	22.058	98.0	1.561
	Fang's FKB	28.985	100	0.0737
	Fang's PKB	18.121	76.8	0.0531
	Gaussian Attack	15.783	0.00	0.00072
$p = 1.0$	The Prop. CMP-FKB-Orgcontr	39.161	100	3.469
	The Prop. CMP-PKB-Orgcontr	33.306	79.8	3.784
	Fang's FKB	31.510	100	0.067
	Fang's PKB	23.306	58.6	0.010
	Gaussian Attack	18.930	0.00	0.00073

Remark: Best Attacking Performance with high time cost

Benchmarks:

- **Arjun' attack** ("Analyzing Federated Learning through an Adversarial Lens", ICML, 2019)

TABLE 6
The attacker's accuracy of the proposed CMP-NKB against Krum aggregation compared with existing works on MLP model and MNIST dataset under various percentages of compromised clients with $p = 0.5$, $U = 50$ and $T = 300$.

Percentage of Compromised Clients	Approach	Attacker's Accuracy (%)
0.20	The Prop. CMP-NKB	75.72
	Arjun's Attack	3.58
	Label Flipping	2.02
0.16	The Prop. CMP-NKB	40.43
	Arjun's Attack	1.30
	Label Flipping	2.15
0.12	The Prop. CMP-NKB	11.59
	Arjun's Attack	2.6
	Label Flipping	5.14
0.08	The Prop. CMP-NKB	12.63
	Arjun's Attack	2.93
	Label Flipping	2.08
0.04	The Prop. CMP-NKB	9.24
	Arjun's Attack	2.80
	Label Flipping	2.21

TABLE 7
The attacker's accuracy of the proposed CMP-NKB against Trimmed mean aggregation compared with existing works on MLP model and MNIST dataset under various degrees of non-i.i.d. with $U = 50$, $M = 10$ and $T = 300$.

The Degree of Non-i.i.d.	Approach	Attacker's Accuracy (%)
$p = 0$	The Prop. CMP-NKB	34.56
	Arjun's Attack	3.67
	Label Flipping	4.72
$p = 0.5$	The Prop. CMP-NKB	64.82
	Arjun's Attack	8.71
	Label Flipping	5.40
$p = 1.0$	The Prop. CMP-NKB	70.23
	Arjun's Attack	10.52
	Label Flipping	5.89

Remark: Best Attacking Performance

Targeted attacking: trains label “9” to label “0”

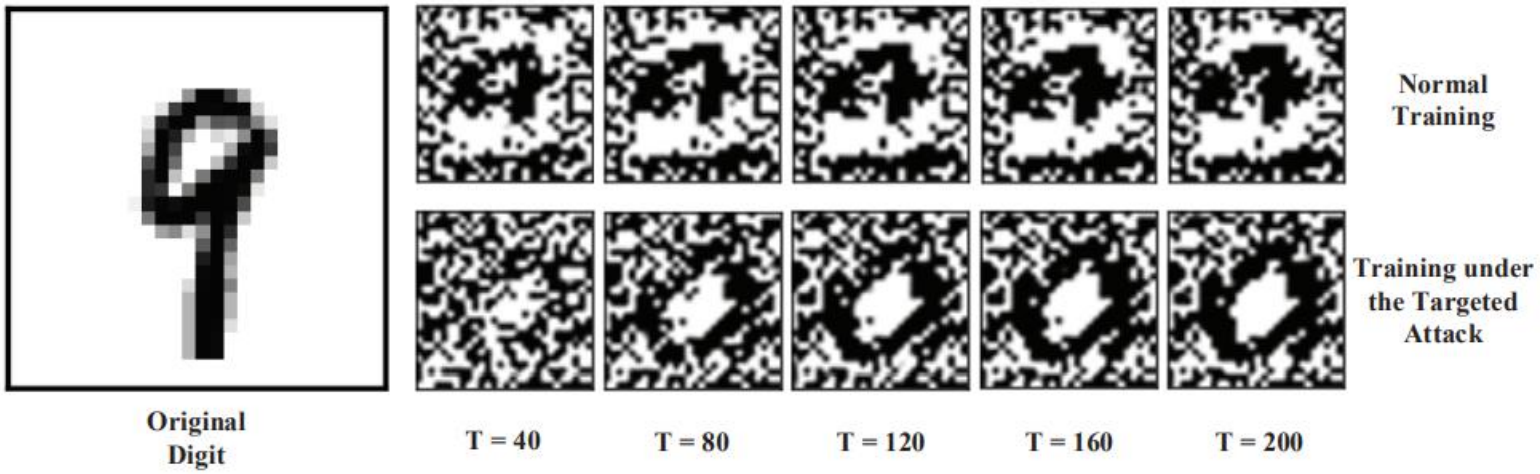
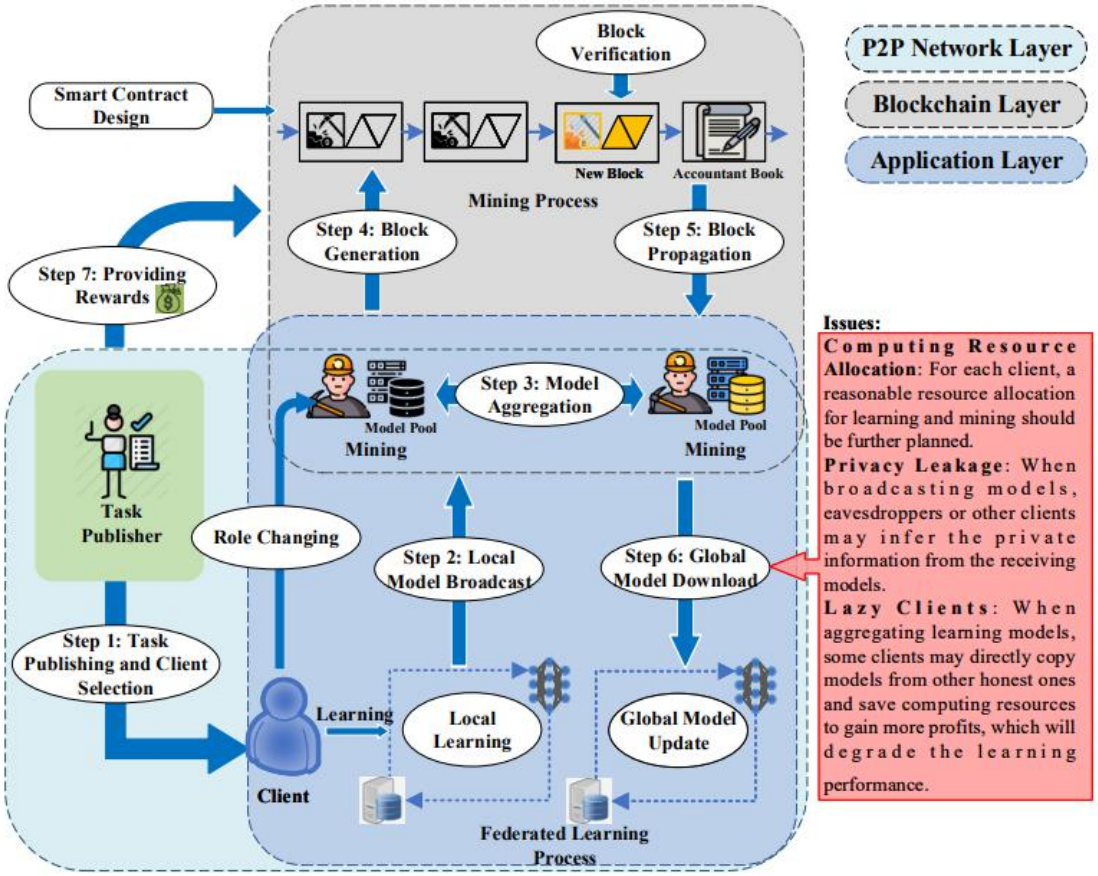


Fig. 4. The visual results of MLP based FL with the interpretability technique corresponding to the digit 9 under our proposed CMP-NKB against Krum aggregation at different communication rounds. Basically speaking, the subfigures show a “typical” 9 understood by the machine model with normal training and targeted attack.

- Wei K, Li J, Ding M, et al. **Covert Model Poisoning Against Federated Learning: Algorithm Design and Optimization**[J]. arXiv preprint arXiv:2101.11799, 2021, submitted to **TDSC**.

Decentralized FL



Advantage:

- Use blockchain to replace the aggeregator
- Participants supervise each other
- Rewards feedback

Issues:

- Resource allocation between mining and training
- Privacy leakage
- Lazy clients: copy the broadcasting models

Ma C, Li J, Ding M, et al. **When Federated Learning Meets Blockchain: A New Distributed Learning Paradigm**[J]. arXiv preprint arXiv:2009.09338, 2020.

Resource allocation

- Li J, Shao Y, Wei K, et al. **Blockchain Assisted Decentralized Federated Learning (BLADE-FL): Performance Analysis and Resource Allocation**[J]. arXiv preprint arXiv:2101.06905, 2021, **submitted TPDS**, under major revision.

Lazy client

- Use **PN sequence** to act as a footprint for each participant

- Warnat-Herresthal S, Schultze H, Shastry K L, et al. **Swarm Learning for decentralized and confidential clinical machine learning[J]**. Nature, 2021, 594(7862): 265-270.

System model:

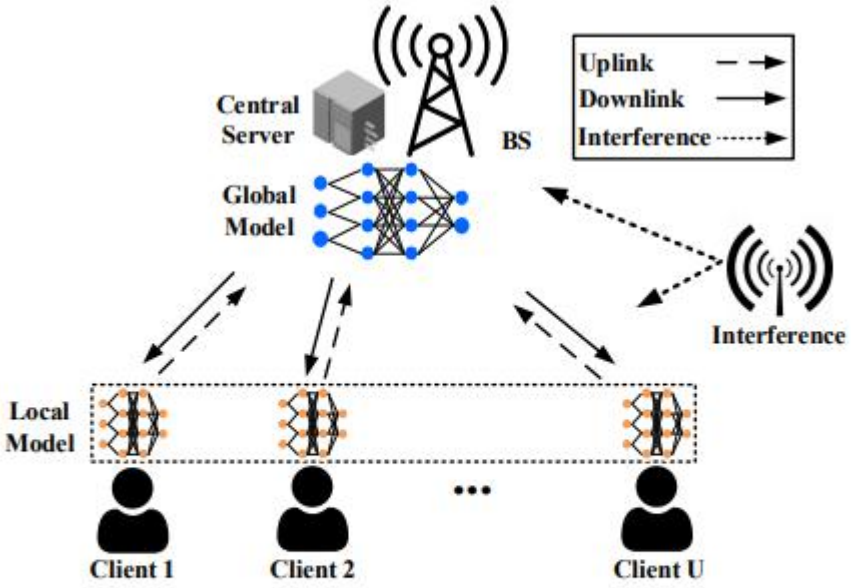


Fig. 1. The framework of wireless FL consists of multiple clients and a BS with multiple channels, where it is non-trivial to characterize the available computing capability for each client and the dynamic wireless channel gains caused by interference.

Problem formulation

Objective: Minimize the total time

$$d_{i,j}(t) = d^{\mathrm{D}}_{i,j}(t) + d^{\mathrm{U}}_{i,j}(t) + d^{\mathrm{L}}_{i}(t).$$

Subject:

- Unknown training resource limitaion
- Privacy requirement
- Uncertian channel conditions

Solution:

Adopt **multi-armed-bandit (MAB)** to optimize the **client scheduling** by leanring statistical property of the interference and computation capability

- Wei K, Li J, Ma C, et al. **Low-Latency Federated Learning over Wireless Channels with Differential Privacy[J]**. arXiv preprint arXiv:2106.13039, 2021, **accepted by JSAC**.

- **Resources limited FL**
  - Limited Communication: Model Compression
  - Limited Computing: Split Learning and Dynamic Neuron Networks
- **Personalized FL**
  - Meta-learning with FL
  - Heterogeneous FL: Non-iid data distribution, Asychorony uploading time, Model diversity
- **Vertical FL**
  - Privacy preserving and Convergence analysis
  - Scaling up participating with aggregation design