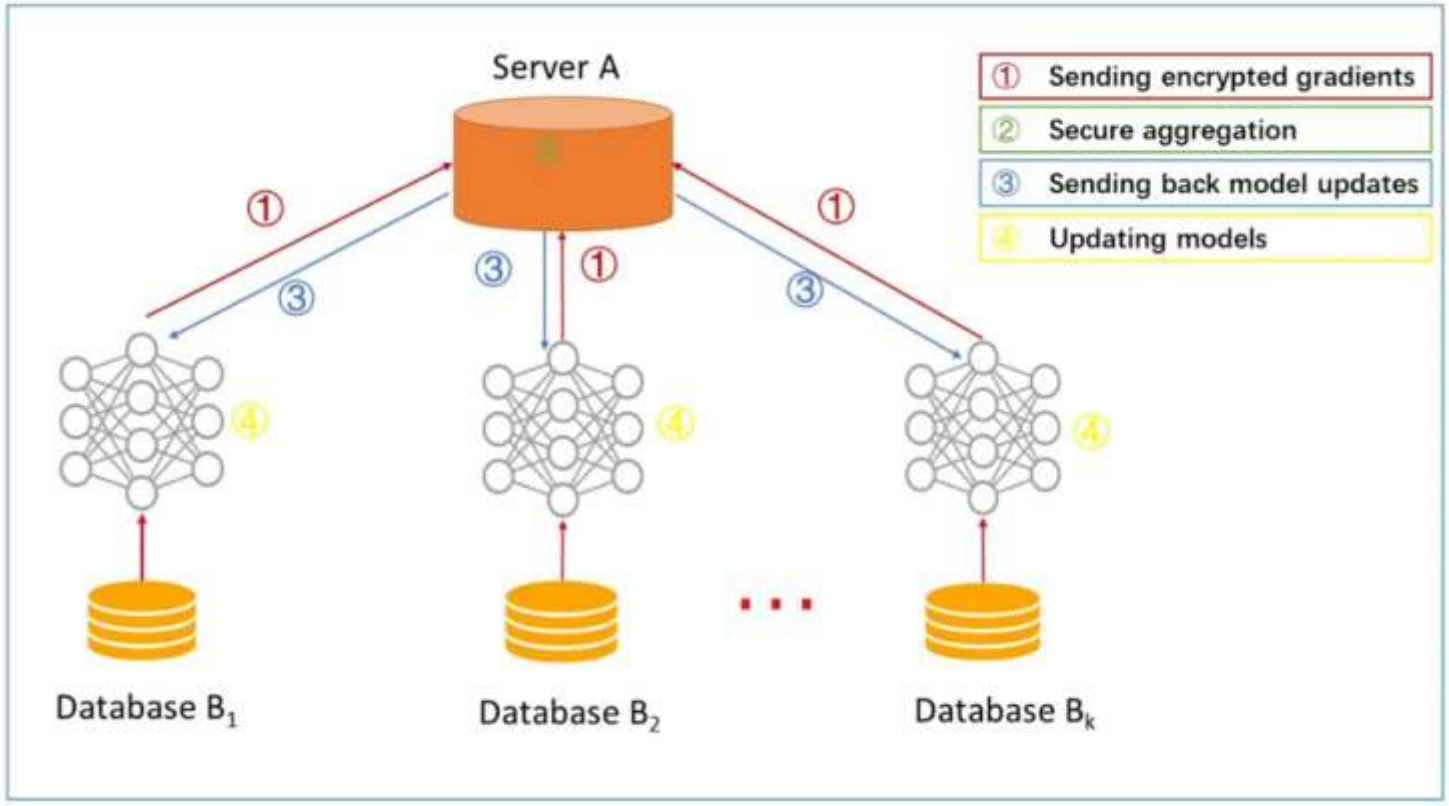
Active Membership Inference Attack under Local Differential Privacy in Federated Learning

2023　CCF C

文章概要

- 设计了一个主动成员推理攻击(AMIA)，由服务器作为攻击者在联邦学习中进行，服务器嵌入恶意参数到全局模型中，推断目标数据样本是否包含在客户端的私有训练数据中；
- 又提出一个在LDP保护下的AMIA；
- 最后在几个基准数据集中评估AMIA的性能。

联邦学习框架



$$G^i = \frac{1}{M}\sum_{j=1}^{M} G_j^i, \quad \theta^{i+1} = \theta^i - \eta G^i$$

AMI威胁模型

$\mathsf{Exp}(\mathcal{A}, \mathcal{L}, \mathbb{D})$:

$\mathcal{D} \sim \mathbb{D}^n$ # Sample n data points from $\mathbb{D}$ into $\mathcal{D}$

$b \xleftarrow{\$} \{0, 1\}$ # Flip a bit b uniformly at random

if $b = 1$ **then**

  $t \xleftarrow{\$} \mathcal{D}$ # Choose t uniformly from $\mathcal{D}$

end

else

  $t \sim \mathbb{D} \setminus \mathcal{D}$ # Sample t from $\mathbb{D}$ s.t. $t \notin \mathcal{D}$

end

$\theta \leftarrow \mathcal{A}^{\mathbb{D}}_{\mathsf{INIT}}(t)$ # The adversary receives t and returns a set of parameters θ

$G \leftarrow \nabla_\theta \mathcal{L}(\mathcal{D}, \theta)$ # Compute the gradients from θ and $\mathcal{D}$

$b' \leftarrow \mathcal{A}^{\mathbb{D}}(t, G)$ # The adversary receives t, G and returns a bit b'

Ret $[b' = b]$ # The game returns 1 if $b' = b$ (the adversary wins), 0 otherwise

Figure 1: AMI Threat Model as a Security Game.

攻击成功率：

$$\begin{aligned}\mathrm{Adv}^{\mathcal{A}} &= \Pr[\mathsf{Exp}(\mathcal{A}, \mathcal{L}, \mathbb{D}) = 1] \\ &= \frac{1}{2}\Pr[b' = 1 | b = 1] + \frac{1}{2}\Pr[b' = 0 | b = 0]\end{aligned}$$

TPR为Pr[b' = 1 | b = 1]

TNR为Pr[b' = 0 | b = 0]

通过梯度推断成员

假设在输入数据点x上，第一个全连接层的输出为ReLU(Wx + b) = max(0, Wx + b)

当$W_i x+b_i \leqslant 0$时，ReLU输出为0，即神经元i没有被x激活，神经元i的梯度$G_i^{(x)}$在数据点x处为0;

如果存在一个神经元i，只被目标数据样本t激活，则有：

$$\begin{cases} \sum_{j=1}^{d} W_{ij} t_j > 0 \\ \sum_{j=1}^{d} W_{ij} x_j \leq 0, \quad \forall x \in \mathcal{D} \setminus t \end{cases}$$

证明：存在W使所有x ≠ t满足上式

令$x_1 = t_1 + c$，$x_i = t_i$，其中 i > 1，c > 0 , w = W_i，有 $w_1 c + \sum_{j=1}^{d} w_j t_j \leq 0 \Longrightarrow -w_1 c \geq \sum_{j=1}^{d} w_j t_j$

同样再令$x_1 = t_1 - c$，$x_i = t_i$，有 $-w_1 c + \sum_{j=1}^{c} w_j t_j \leq 0 \Longrightarrow w_1 c \geq \sum_{j=1}^{d} w_j t_j$

则不存在W使所有x ≠ t满足上式

改进

在第二个完全连接层选择一个神经元，使得该神经元仅被目标数据样本激活，将h表示为第二层所选神经元的权重向量，存在(h, W)，则攻击成功，有：

$$\begin{cases} \sum_{i=1}^{r} h_i \operatorname{ReLU}(\sum_{j=1}^{d} W_{ij} t_j) > 0 \\ \sum_{i=1}^{r} h_i \operatorname{ReLU}(\sum_{j=1}^{d} W_{ij} x_j) \leq 0, \quad \forall x \neq t \end{cases}$$

训练所选神经元

在神经元的输出上提出一个逻辑s型函数σ，

使用交叉熵损失训练选择的神经元，

试图使s(t) = 1, 有hReLU(Wt) > 0;

s(x) = 0，有hReLU(Wx) < 0

$$\begin{aligned} s(x) &= \sigma(\sum_{i=1}^{r} h_i \operatorname{ReLU}(\sum_{j=1}^{d} W_{ij} x_j)) \\ &= \sigma(h \cdot \operatorname{ReLU}(Wx)) \end{aligned}$$

攻击策略

adversary $\mathcal{A}^{\mathbb{D}}_{\mathsf{INIT}}(t)$:

$\mathcal{X} \sim \mathbb{D}^m$

$D_A \leftarrow \bigcup_{x \in \mathcal{X} \setminus t} \{(x, 0)\}$

$D_A \leftarrow D_A \cup \{(t, 1)\}$

Train h, W (Eq. 6) from dataset D_A

Initialize θ

$\theta \leftarrow \theta \cup (h, W)$

Ret θ

adversary $\mathcal{A}^{\mathbb{D}}(t, G)$:

Extract g_t as the gradient of the chosen neuron from G

Ret $[g_t \neq 0]$

Figure 2: AMI Attack Strategy of the Adversary $\mathcal{A}$.

LDP下的AMI威胁模型

$\underline{\mathsf{Exp}_{LDP}(\mathcal{A}_{LDP}, \mathcal{L}, \mathbb{D}, \mathcal{M}, \varepsilon):}$
$\mathcal{D} \sim \mathbb{D}^n$
$b \xleftarrow{\$} \{0, 1\}$
if $b = 1$ **then**
  $t \xleftarrow{\$} \mathcal{D}$
end
else
  $t \sim \mathbb{D} \setminus \mathcal{D}$
end
$\theta \leftarrow \mathcal{A}^{\mathbb{D},\mathcal{M}}_{LDP,\mathsf{INIT}}(t, \varepsilon)$
$\mathcal{D}' \leftarrow \mathcal{M}(\mathcal{D}, \varepsilon)$ # Apply the LDP mechanism on $\mathcal{D}$
$G \leftarrow \nabla_\theta \mathcal{L}(\mathcal{D}', \theta)$
$b' \leftarrow \mathcal{A}^{\mathbb{D},\mathcal{M}}_{LDP}(t, G, \varepsilon)$
Ret $[b' = b]$

Figure 3: AMI Threat Model under LDP Mechanisms.

攻击策略

$$\begin{cases} h \cdot \mathrm{ReLU}(W\mathcal{M}(t,\varepsilon)) > 0 \\ h \cdot \mathrm{ReLU}(Wx) \leq 0, \quad \forall x \neq \mathcal{M}(t,\varepsilon) \end{cases}$$

adversary $\mathcal{A}^{\mathbb{D},\mathcal{M}}_{LDP,\mathsf{INIT}}(t,\varepsilon)$:

Choose $l \in \mathbb{N}$
$\mathcal{T} \leftarrow \emptyset$
for $i = 1$ *to* l **do**
  $\mathcal{T} \leftarrow \mathcal{T} \cup \{\mathcal{M}(t,\varepsilon)\}$
end
$\mathcal{X} \sim \mathbb{D}^m \setminus \mathcal{T}$ # Sample $\mathcal{X} \sim \mathbb{D}^m$ s.t. $\mathcal{X} \cap \mathcal{T} = \emptyset$
$D_A \leftarrow \bigcup_{x \in \mathcal{X}} \{(x,0)\}$
$D_A \leftarrow D_A \cup \left(\bigcup_{x \in \mathcal{T}} \{(x,1)\}\right)$
Train h, W (Eq. 6) from dataset D_A
Initialize θ
$\theta \leftarrow \theta \cup (h, W)$
Ret θ

adversary $\mathcal{A}^{\mathbb{D},\mathcal{M}}_{LDP}(t,G,\varepsilon)$:

Extract g_t as the gradient of the chosen neuron from G
Ret $[g_t \neq 0]$

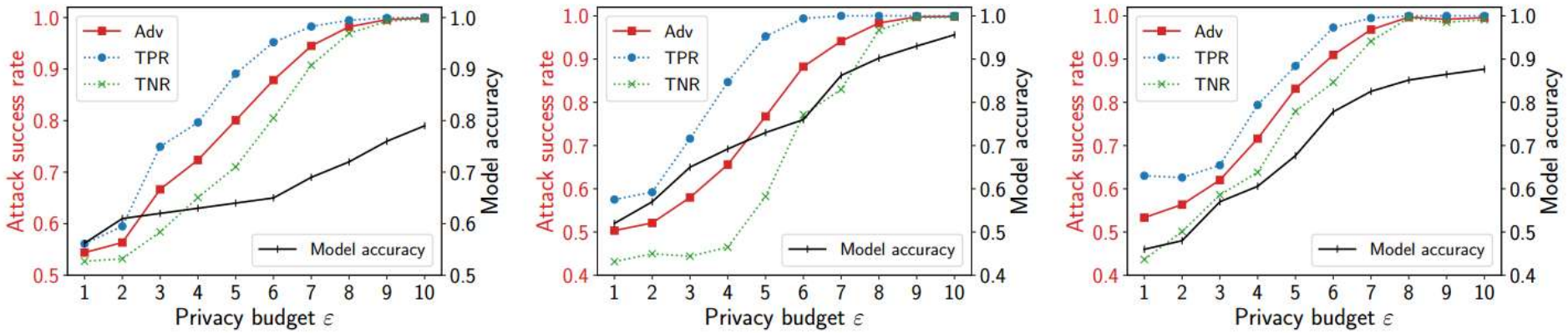
Figure 4: Attack Strategy of the Adversary $\mathcal{A}_{LDP}$.

实验

- LDP下的攻击性能

TPR为Pr[b' = 1 | b = 1]

TNR为Pr[b' = 0 | b = 0]

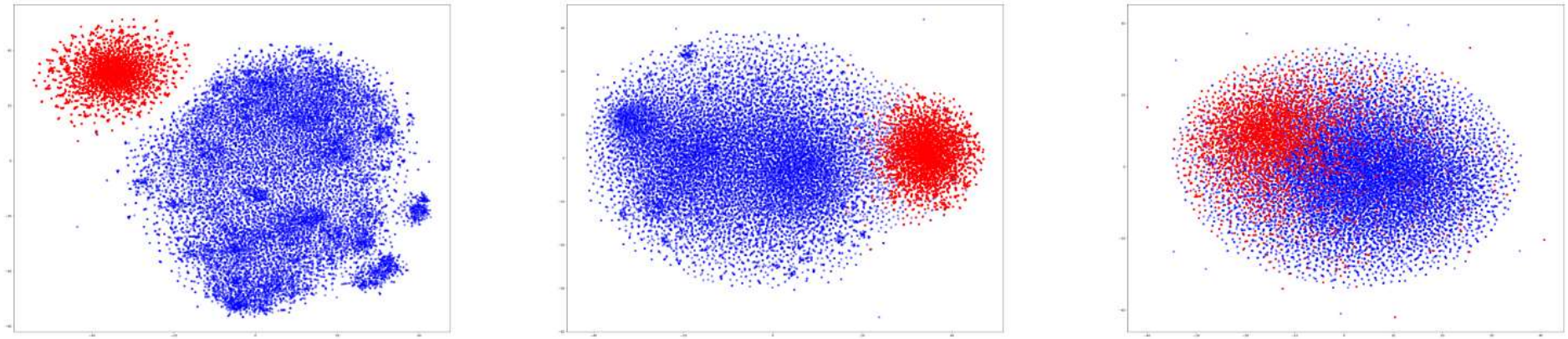


(a) CelebA (b) ImageNet (c) CIFAR-10

Figure 5: Attack success rate of AMI against an ε-LDP mechanism on CelebA, ImageNet, and CIFAR-10 datasets. The success rate is represented via the advantage (Adv), true positive rate (TPR), and true negative rate (TNR) according to Eq. 3. The baseline of random guessing is 0.5. The model accuracy illustrates the utility loss of the data when using LDP.

LDP下训练选择的神经元

使用t-SNE算法可视化了训练集D中的样本分布



(a) $\varepsilon = 10$ (b) $\varepsilon = 7$ (c) $\varepsilon = 5$

Figure 6: Visualizing the distribution of the target sample t among other samples in the training set $\mathcal{D}$ using t-SNE embeddings. The red dots denote the target sample t and a multitude of its LDP noises $\mathcal{M}(t, \varepsilon)$, while the blue dots denote other non-target samples. These data samples are obtained from the CelebA dataset.