GFL: Federated Learning on Non-IID Data via Privacy-preserving Synthetic Data

汇报人：黄其涵　　导师：章静教授

2023 IEEE International Conference on Pervasive Computing and Communications (PerCom)

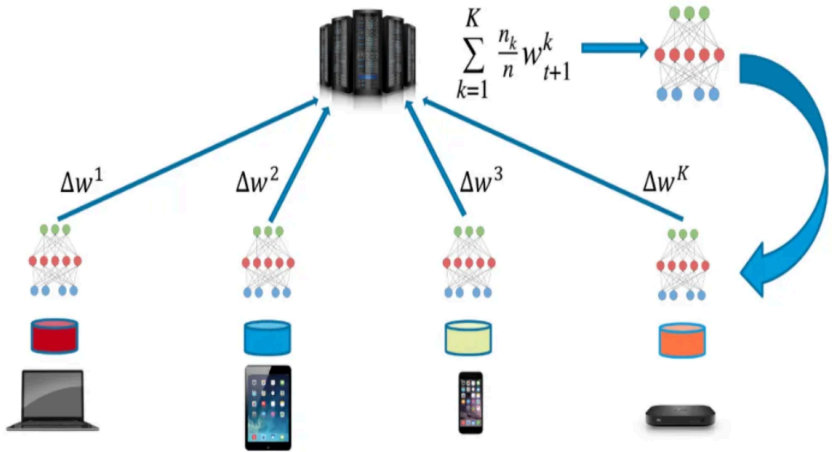
June 8, 2023

目录

1.1 Federated Learning (联邦学习)

- 联邦学习是一种分布式机器学习方法, 其利用一个中央服务器 (也称为服务器端) 协调各终端设备 (也称为客户端), 协同训练一个各客户端共享的全局模型。
- 与传统中心化训练方法不同, **联邦学习不需要各设备发送自身隐私数据至数据中心, 因此有利于保护数据隐私**。具体而言, 联邦学习在客户端和服务器端之间通过多轮通信迭代优化模型。
- 每轮通信包含两个阶段:
  (1) 各客户端从服务器端下载全局模型, 并在本地数据上进行训练以获得本地模型;
  (2) 服务器端接收并聚合各客户端的本地模型参数以获得性能更优的全局模型. 然而, 现有联邦学习机制尚面临两大不足。

1.1 Federated Learning (联邦学习)



1.2 现有问题

联邦学习的现存问题：

- **类别不均衡**。全局模型需考虑多个客户端的数据，但各客户端往往仅包含部分类别数据且类别间数据量严重不均衡，使得全局模型难以训练。
- **数据分布差异**。由于各客户端的功能和用户使用习惯不同，不同客户端往往产生不同类别的数据，导致各客户端数据之间的类别分布差异较大。

在这项工作中，本文旨在通过提出一种基于 GAN 的方法来解决高度倾斜的数据分布问题，同时保护客户的隐私，从而构建高性能的全局模型。**而基于 GAN 技术的联邦学习方法的主要挑战如下：**

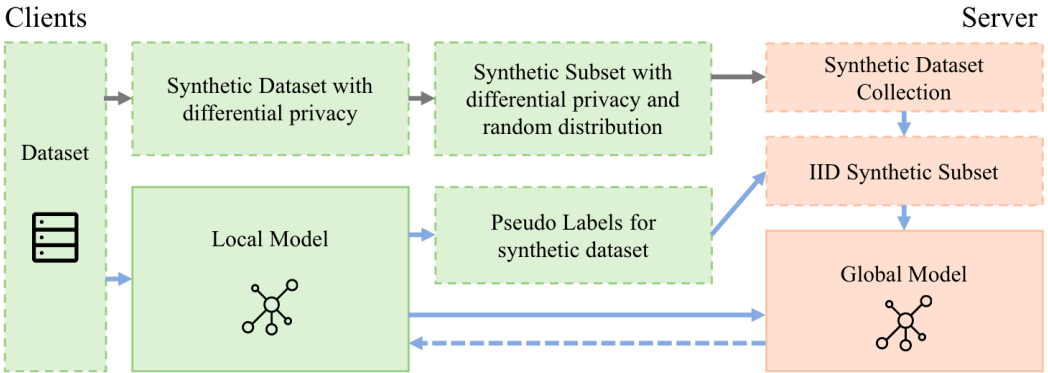
- **如何在保护本地客户隐私的同时共享合成数据？**
- **如何在服务器上正确使用从客户端收集的合成数据？**
- **如何应对 GAN 引入的潜在模式崩溃和低质量数据？**

1.3 主要贡献

- **本文提出基于 GAN 的 GFL 框架以解决 FL 中数据异构性问题，并为 FL 系统构建高性能全局模型**。GFL 促进了 FL 训练过程，同时保持了成员身份的隐私和原始数据在客户中的分布。
- **本文提出了一种保护隐私的数据生成工作流程，以生成符合我们隐私设置的合成样本**。 DPGAN 用于生成满足差分隐私的样本，以保护客户端免受成员推理攻击。本文生成大量合成数据并选择一个随机分布的子集来隐藏客户端的真实数据分布。
- **本文对合成数据进行 “iidify” 处理，以 iid 方式训练全局模型，并使用 Epoch Decay 参数来解决模式崩溃问题并避免使用低质量的合成数据**。

目录

3.1 Overview of Generative Federated Learning (GFL)



3.2 Synthetic Data Generation

Threat Model

将 GAN 引入 GFL 中的联邦学习。但是，GAN 提供的合成数据可能会泄露客户的隐私信息。**GAN 中最严重的漏洞之一是成员关系问题。例如，即使他/她参加了 GAN 训练，也没有人希望他/她的脸出现在生成的图像中**。

本文将考虑虚拟一个对手（GFL 中的服务器），他的目的是**推断一个单一的已知记录是否包括在受害者（目标 GAN 模型）的训练集中，这被称为成员推理攻击**。对手是半诚实的，他遵循正常的训练过程，但对成员关系很好奇。作为对手的半诚实服务器处于如下所述的黑盒设置中。

- The architecture of the victim.
- The parameters and training hyper-parameters of the victim.
- The dataset used in the victim.

3.2 Synthetic Data Generation

Privacy-preserving Data Generation

服务器作为协调整个 FL 训练的角色，本文希望它能够主动参与训练，而不是被动地聚合参数。由于缺少特征信息，我们想为服务器创建一个小型 iid 数据集，以帮助在短时间内找到全局最优值。这个 iid 数据集中的数据应该完全来自客户端，所以仍然面临着共享时数据隐私信息可能泄露的挑战。

本文专注于抵抗成员推理攻击和防止数据分布信息泄漏，在合成数据生成阶段，提供了一种实现合成数据隐私保护生成的解决方案。

3.2 Synthetic Data Generation

Privacy-preserving Data Generation

Algorithm 2: Privacy-preserving Data Generation

Input : The synthetic data amount n_{S_k}.

Output: The synthetic dataset S_k.

Data : The local dataset $\mathcal{P}_k$ with size n_k.

Client k **executes:**
- $P_{S_k} \leftarrow$ (a random distribution)
- $d_k, g_k \leftarrow \text{DPGANTraining}(\mathcal{P}_k)$
- $S'_k \leftarrow \emptyset$
- **while** *not* $\text{EnoughData}(S'_k, n_{S_k}, P_{S_k})$ **do**
  - $noises \leftarrow$ (random noises)
  - $S'_k \leftarrow S'_k \cup g_k(noises)$
- $S_k \leftarrow \text{CutData}(S'_k, n_{S_k}, P_{S_k})$

EnoughData(S'_k, n_{S_k}, P_{S_k}**):**
- **for** *each label* i **do**
  - **if** $S'^i_k < n_{S_k} \cdot P^i_{S_k}$ **then**
    - **return** *false*
- **return** *true*

CutData(S'_k, n_{S_k}, P_{S_k}**):**
- $S_k \leftarrow \emptyset$
- **for** *each label* i **do**
  - $temp \leftarrow$ (a random subset of S'^i_k with the size of $n_{S_k} \cdot P^i_{S_k}$)
  - $S_k \leftarrow S_k \cup temp$
- **return** S_k

3.3 Federated Learning with Synthetic Data

Algorithm 3: Online Federated Learning Stage of GFL

Input : The number of communication rounds T, the number of total clients K, the proportion of selected clients ϕ, the number of server epochs E_s, and the epoch decay parameter τ.

Output: The final global model w_T.

Data : The local dataset $\mathcal{P}_k$ with size n_k and the synthetic dataset S_k with size n_{S_k} on each client $k = 1, 2, \cdots, K$.

Server executes:
  $S' \leftarrow \{S_1, S_2, \cdots, S_K\}$
  initialize w_0
  for *each round* $t = 1, 2, \cdots, T$ **do**
    $m \leftarrow \max(\phi \cdot K, 1)$
    $C_t \leftarrow$ (random subset of m clients)
    for *each client* $k \in C_t$ **do**
      $y_{S_k}, w_t^k \leftarrow \text{ClientUpdate}(k, w_{t-1})$
    $w_t' \leftarrow \sum_{k=1}^{m} \frac{n_k}{n} \cdot w_t^k$
    update labels of S' with $\{y_{S_k}, k \in C_t\}$
    $S \leftarrow \text{IIDify}(S')$
    $w_t \leftarrow \text{ServerUpdate}(t, w_t')$
  return w_T

IIDify(S'):
  $mini \leftarrow \min\{|S'^i|, i = 1, 2, \cdots, \mathcal{Y}\}$
  $S \leftarrow \emptyset$
  for *each label* i **do**
    $temp \leftarrow$ (a random subset of S'^i with the size of $mini$)
    $S \leftarrow S \cup temp$
  return S

ClientUpdate(k, w):
  ... (similar to ClientUpdate in Algorithm 1)
  $y_{S_k} \leftarrow w(S_k)$
  return y_{S_k}, w

ServerUpdate(t, w):
  $E \leftarrow E_s \cdot e^{-\tau \cdot (t-1)}$
  ... (similar to ClientUpdate in Algorithm 1)
  return w

目录

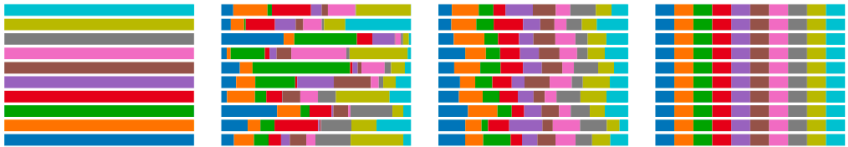
4.1 数据配置
数据集

在我们的实验中，我们使用 DPGAN 作为 Local GAN 模型，ResNet18 作为 EMNIST 和 CIFAR-10 联邦学习的训练模型。

1) **EMNIST**。在实验中使用它的 “Digits” 子集。它包含 10 个类，总共 280, 000 个样本，分为 240, 000 个训练样本和 40, 000 个测试样本。与 MNIST 相比，它拥有更多数据并且完全平衡。
2) **CIFAR-10**。它包含 10 个不同类别的 60, 000 个 32×32 彩色图像，与 EMNIST 相比，这是一项更复杂的任务。我们使用这个数据集来衡量 GFL 在艰巨任务中的稳定性。

4.2 数据配置

Dirichlet Distribution



(a) $\alpha \to 0$ (b) $\alpha \to 1$ (c) $\alpha \to 10$ (d) $\alpha \to \infty$

Fig. 2. Data distributions of non-iid clients sampled from different Dirichlet distributions. Different colors represent different labels, and each row is the distribution of a client. (a) $\alpha \to 0$. An extreme situation when each client holds data in only one category. (b) $\alpha \to 1$. A common simulated non-iid situation. (c) $\alpha \to 10$. A slightly non-iid situation. (d) $\alpha \to \infty$. IID distributions among all clients.

4.2 对比模型

本文将 GFL 与目前最新的方法进行比较, 即 **FedAvg, FedPoxr, SCAFFOLD 和 FedNvao**。本实验通过 A 狄利克雷分布 Dir(0.1) 模拟各客户端数据的类别分布。

GFL 在合成数据生成阶段训练了 200 个 epoch。然后，我们从每个客户端 k 中随机选择 $n_{S_k} = 500$ 个合成样本。在联邦学习阶段，初始全局训练 epoch E_s 设置为 10，如果未指定，epoch decay 参数 τ 设置为 0.1。

上述所有算法共享相同的基本训练设置。训练包含 100 轮通信，每轮每个客户端训练 20 个 epoch，服务器和客户端的批大小都设置为 128。

4.3 实验结果
Result Analysis

TABLE II
THE ACCURACY RESULTS OF DIFFERENT METHODS ON EMNIST AND CIFAR-10. GFL IS THE BEST IN MOST SITUATIONS, ESPECIALLY WHEN THE DATA IS EXTREMELY NON-IID SUCH AS $\alpha \to 0, 1$. AS THE DATA DISTRIBUTION MOVES TOWARDS IID, GFL MAINTAINS THE ACCURACY SIMILAR TO FEDAVG.

EMNIST	$\alpha \to 0$		$\alpha \to 1$		$\alpha \to 10$		$\alpha \to \infty$	
FedAVG	30.84%	(x1.00)	66.05%	(x1.00)	98.55%	(x1.00)	98.65%	(x1.00)
FedProx	64.72%	(x2.10)	77.66%	(x1.18)	98.94%	(x1.00)	**98.89%**	**(x1.00)**
SCAFFOLD	51.35%	(x1.67)	78.38%	(x1.19)	97.88%	(x0.99)	98.03%	(x0.99)
GFL	**78.78%**	**(x2.55)**	**89.77%**	**(x1.36)**	**99.39%**	**(x1.01)**	98.13%	(x0.99)

CIFAR-10	$\alpha \to 0$		$\alpha \to 1$		$\alpha \to 10$		$\alpha \to \infty$	
FedAVG	13.43%	(x1.00)	46.67%	(x1.00)	78.15%	(x1.00)	89.84%	(x1.00)
FedProx	38.55%	(x2.87)	46.19%	(x0.99)	83.63%	(x1.07)	89.31%	(x0.99)
SCAFFOLD	38.90%	(x2.90)	56.82%	(x1.22)	**88.50%**	**(x1.13)**	89.48%	(x1.00)
GFL	**56.22%**	**(x4.19)**	**67.42%**	**(x1.44)**	84.26%	(x1.08)	**89.93%**	**(x1.00)**

1. GFL 在所有设置中都优于其他算法; 2. non-iid 程度越大，GFL 的优势越明显;
3. 数据集越复杂，GFL 越优越。

4.3 实验结果
Result Analysis

TABLE III
ACCURACY COMPARISON OF FEDAVG AND GFL WITH DIFFERENT EPOCH DECAY PARAMETER τ.

CIFAR-10	τ	$\alpha \to 0$	$\alpha \to 1$
FedAVG	∞	13.43%	46.67%
GFL	0.1	**56.22%**	**67.42%**
	0	32.77%	38.21%
	$-\infty$	17.26%	24.94%

1. Epoch Delay 参数在 GFL 中是必不可少的;
2. 只使用合成数据的方法是没有用的。

4.3 实验结果
Result Analysis

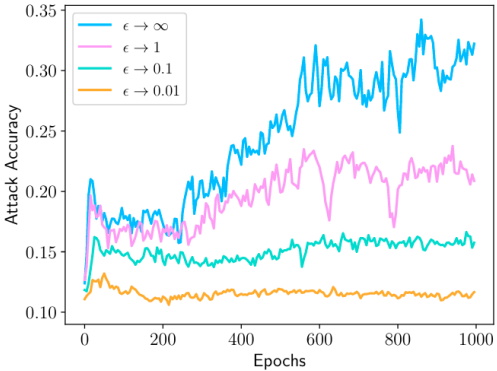


Fig. 3. The line chart of the privacy evaluation results on different privacy level ϵ. The membership inference attack accuracy drops as ϵ decreases.

4.3 实验结果
Result Analysis

TABLE IV
THE TABLE OF THE PRIVACY EVALUATION AND FL TRAINING RESULTS ON DIFFERENT PRIVACY LEVEL ϵ.

CIFAR-10	ϵ	Attack Acc	FL Acc
GFL	∞	31.16%	69.31%
	1	21.58%	68.12%
	0.1	14.80%	67.42%
	0.01	11.66%	52.79%
FedAVG	—	—	46.67%

根据图 3 和表 IV 中的实验，选择 $\epsilon = 0.1$，这可以防止成员推理攻击，同时保持良好的性能。

目录

研究总结

本文提出了一个称为生成联合学习 (GFL) 的新框架来处理非独立同分布问题。我们将 DPGAN 引入 GFL 以应对潜在的隐私攻击，尤其是成员推理攻击。

大量实验表明，GFL 可以在非 iid 情况下工作，也可以在 iid 情况下工作，并且在大多数情况下优于其他框架，如 FedProx 和 SCAFFOLD。隐私评估表明，通过在 DPGAN 的训练中加入适当的噪声，我们可以有效地防御成员推理攻击。我们还分析了 GFL 引入的 Epoch Decay 参数的必要性，这对于防止模式崩溃和低质量样本损害全局模型很重要。

Q&A

感谢您的聆听和反馈