基于本地差分隐私的异步横向联邦安全梯度聚合方案

魏立斐[①②] 张无忌[①] 张 蕾[*①] 胡雪晖[③] 王绪安[④]

[①](上海海洋大学信息学院 上海 201306)

[②](上海海事大学信息工程学院 上海 201306)

[③](上海同态信息科技有限责任公司 上海 200235)

[④](武警工程大学 陕西 西安 710086)

摘 要：联邦学习作为一种新兴的分布式机器学习框架，通过在用户私有数据不出域的情况下进行联合建模训练，有效地解决了传统机器学习中的数据孤岛和隐私泄露问题。然而，联邦学习存在着训练滞后的客户端拖累全局训练速度的问题，异步联邦学习允许用户在本地完成模型更新后立即上传到服务端并参与到聚合任务中，而无需等待其他用户训练完成。然而，异步联邦学习也存在着无法识别恶意用户上传的错误模型，以及泄露用户隐私的问题。针对这些问题，该文设计一种面向隐私保护的异步联邦的安全梯度聚合方案(SAFL)。用户采用本地差分隐私策略，对本地训练的模型添加扰动并上传到服务端，服务端通过投毒检测算法剔除恶意用户，以实现安全聚合(SA)。最后，理论分析和实验表明在异步联邦学习的场景下，提出的方案能够有效识别出恶意用户，保护用户的本地模型隐私，减少隐私泄露风险，并相对于其他方案在模型的准确率上有较大的提升。

关键词：安全聚合；本地差分隐私；隐私保护；恶意投毒攻击；异步联邦学习

中图分类号：TN919; TP309 **文献标识码**：A **文章编号**：1009-5896(2024)07-3010-09

DOI: 10.11999/JEIT230923

A Secure Gradient Aggregation Scheme Based on Local Differential Privacy in Asynchronous Horizontal Federated Learning

WEI Lifei[①②] ZHANG Wuji[①] ZHANG Lei[①] HU Xuehui[③] WANG Xuan[④]

[①](*College of Information Technology*, *Shanghai Ocean University*, *Shanghai* 201306, *China*)

[②](*College of Information Engineering*, *Shanghai Maritime University*, *Shanghai* 201306, *China*)

[③](*Shanghai Tongtai Information Technology Co., Ltd.*, *Shanghai* 200235, *China*)

[④](*Engineering University of PAP*, *Xi'an Shaanxi* 710086, *China*)

Abstract: Federated learning is an emerging distributed machine learning framework that effectively solves the problems of data silos and privacy leakage in traditional machine learning by performing joint modeling training without leaving the user's private data out of the domain. However, federated learning suffers from the problem of training-lagged clients dragging down the global training speed. Related research has proposed asynchronous federated learning, which allows the users to upload to the server and participate in the aggregation task as soon as they finish updating their models locally, without waiting for the other users. However, asynchronous federated learning also suffers from the inability to recognize malicious models uploaded by malicious users and the problem of leaking user's privacy. To address these issues, a privacy-preserving Secure Aggregation scheme for asynchronous Federated Learning(SAFL) is designed. The users add perturbations to locally trained models and upload the perturbed models to the server. The server detects and rejects the malicious users through a poisoning detection algorithm to achieve Secure Aggregation(SA). Finally, theoretical analysis and experiments show that in the scenario of asynchronous federated learning, the proposed

收稿日期：2023-08-28；改回日期：2023-12-21；网络出版：2023-12-26

*通信作者： 张蕾 Lzhang@shou.edu.cn

基金项目：国家自然科学基金(61972241, 62172436)，上海市自然科学基金(22ZR1427100)，陕西省自然科学基金(2023-JC-YB-584)，上海市软科学研究项目(23692106700)

Foundation Items: The National Natural Science Foundation of China (61972241, 62172436), The Natural Science Foundation of Shanghai (22ZR1427100), The Natural Science Foundation of Shaanxi Province (2023-JC-YB-584), The Soft Science Project of Shanghai (23692106700)

scheme can effectively detect malicious users while protecting the privacy of users' local models and reducing the risk of privacy leakage. The proposed scheme has also a significant improvement in the accuracy of the model compared with other schemes.

Key words: Secure Aggregation (SA); Local differential privacy; Privacy preserving; Malicious poisoning attack; Asynchronous federated learning

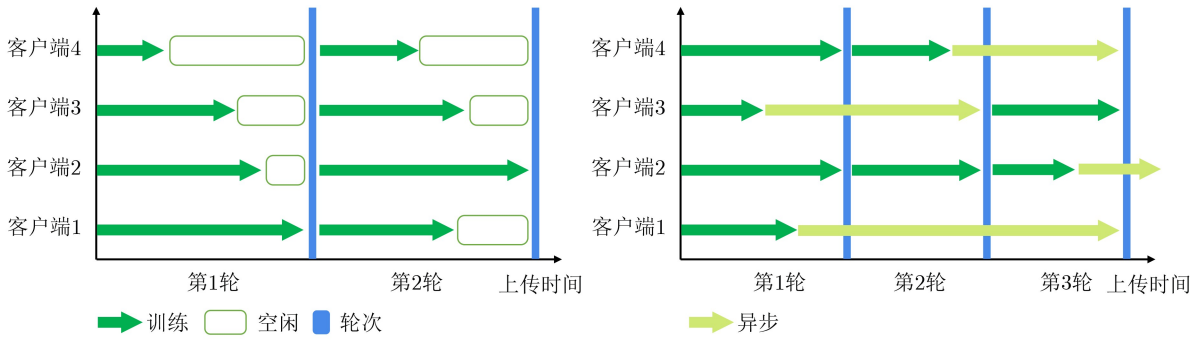
1　引言

近年来，随着大数据和分布式数据挖掘的兴起，机器学习、人工智能等技术的不断发展，基于机器学习的数据挖掘技术被广泛的应用。传统的机器学习场景中，数据所有者需要将本地数据发送给中央服务器进行统一建模训练，可能会遇到参与方与中心节点之间的网络连接慢、通信量延迟高以及模型数据被第三方窃取的问题。此外，当前所需要的训练数据往往分散在不同的用户以及行业寡头手中，导致了数据孤岛的存在。为了应对这些问题，谷歌公司于2016年首次提出了联邦学习框架的概念[1]，极大地解决了多个手机客户端中进行模型更新时通信效率低下以及数据孤岛问题。根据不同的应用场景，可以将联邦学习分为横向联邦学习、纵向联邦学习以及联邦迁移学习[2]。横向联邦学习的各个参与方的本地数据具有相似的特征和分类标签，是目前主流的联邦学习形式之一。然而，传统的横向联邦学习中存在一个同步训练延迟[3]的问题。由于中央服务器必须等待当前所有参与方完成其本地模型更新后才能进行全局模型的更新，可能出现某些客户端训练速度慢或延迟高的情况，从而拖累了全局训练的进度。

为了解决横向联邦学习中的训练延迟问题，研究人员提出了加速方式——异步联邦学习[4]方法。异步联邦学习采用更加灵活的方式，通过使用缓冲区接收异步发送来的模型更新，从而实现动态更新全局模型。图1展示了异步联邦学习和传统的同步联邦学习在训练方式上的不同。Liu等人[5]提出了一种自适应的异步联邦学习方式，允许参与方在本地完成模型更新后立即上传到服务端并参与到聚合任务中，而无需等待其他参与方是否完成本地训练。文献[6]的研究通过高并发实验证明了异步联邦学习的训练速度相比同步联邦学习提高了5倍，通信开销减少到原来的1/8。在异步联邦学习中，服务端可以通过衰减策略[4]来确定参与方上传模型在聚合中的权重。具体而言，服务端根据参与方上传模型的时间先后顺序，为每个机器学习模型分配不同的聚合权重。这样能够很好的平衡各个参与方之间的贡献，适应不同参与方的计算能力和数据分布，从而提升全局模型的训练速度，达到优化整体学习效果的目的。

然而，相关研究表明，异步联邦学习同样存在安全漏洞问题[7, 8]，尤其是在模型中毒和隐私泄露[9]方面。攻击者可以利用这些漏洞破坏数据隐私性，分为以下3个方面：首先，在训练模型阶段中，由于参与方仅上传模型参数，中央服务器无法确定这些模型参数是否具有恶意性。这种情况容易导致聚合后的全局模型中融合恶意模型，进而影响模型的收敛性。其次，在异步联邦学习中，攻击者可能会出于破坏的目的，积极上传恶意模型。由于采用了以上传模型时间为优先级的异步聚合机制，这些恶意模型将获得更高的聚合权重，进一步加剧了模型投毒攻击的危害。同时，攻击者存在一定几率能够通过训练模型中的信息反推出参与方的训练数据，这将破坏系统的机密性；或通过分析模型参数或模型输出，推断出参与方的敏感数据，破坏系统的隐私性和安全性。

针对异步联邦学习的安全问题来说。So等人[10]



(a) 同步联邦学习　　(b) 异步联邦学习

图 1 异步联邦学习与传统同步联邦学习对比图

提出一种轻量化异步联邦学习方法，用户之间使用秘密共享两两配对生成掩码，以保护本地模型隐私，防止被服务端窃取数据。Fang等人[11]设计了一种稳健的异步联邦学习方法，通过计算客户端模型与服务器模型在更新方向以及幅度上的差距，选择拒绝或者聚合该客户端模型，用于抵御拜占庭式数据投毒攻击。Wang等人[12]基于区块链的身份管理授权机制，将异步联邦学习框架部署在区块链上，在实现保护隐私的同时加快训练过程。Lu等人[13]使用一种点对点更新模型的异步聚合方案，结合本地差分隐私带来比集中式更高的模型安全性，但是仍难以抵御数据投毒攻击。Damaskinos等人[14]提出的方案虽然可以抵御投毒攻击，但是需要服务端缓存一些训练样本，这种做法可能会引发中央服务器端的隐私泄露问题。总体来说，这些采用异步联邦学习方案的研究没有充分考虑防御投毒攻击和保护模型隐私这两个关键的安全问题。

针对上述安全问题，现有研究中采用同态加密[15–18]、安全多方计算[19–21]等技术来保护模型的隐私性。然而，这些方案都需要较高的计算能力和通信开销，影响异步联邦学习的训练性能。为此，本文提出一种基于本地差分隐私的异步联邦的安全梯度聚合方案(Secure Aggregation scheme for asynchronous Federated Learning, SAFL)，主要贡献如下：(1)为了解决攻击者上传恶意模型的问题，提出一种检测恶意用户的安全梯度聚合方法，能够有效降低投毒模型对全局模型的干扰，提高训练模型的鲁棒性。(2)针对模型隐私泄露问题，引入本地差分隐私技术来增强模型的隐私性，通过对参与方的本地模型添加噪声来混淆敏感信息，有效减少个体数据隐私的泄露风险。(3)通过实验评估了在不同场景和不同数量恶意用户下的聚合效果，该结果证实了该方法可以有效检测出恶意用户，并具有较高的模型准确率。各个异步联邦学习方案在抵御投毒攻击和隐私保护方面的差异如表1所示。

2 安全的异步联邦学习方案

本文假设服务端和大部分参与者都是半诚实的，会按照协议要求进行训练，但他们可能存在收集和提取全局模型中敏感信息的风险。恶意攻击者可以控制一部分的客户端，上传恶意模型，使其偏离正常更新方向，甚至阻止全局模型收敛。在每轮的全局训练中，攻击者以一定的概率被选中参与本轮训练，干扰全局模型的更新过程。同时，攻击者接受服务端的调度，但不清楚训练所使用的聚合算法。

表 1 异步联邦学习方案对比

相关方法	抵御投毒攻击	客户端的隐私保护	服务端的隐私保护
文献[10]	不支持	不支持	支持
文献[11]	支持	不支持	不支持
文献[12]	不支持	支持	支持
文献[13]	不支持	支持	支持
文献[14]	支持	不支持	不支持
本文	支持	支持	支持

首先，服务端随机选择客户端并行化参与训练，并异步接收参与方发送来的训练模型(如算法1所示)。其次，客户端同时进行本地模型训练和模型预测准确率评估的任务(如算法2所示)。最后，服务端收到客户端的更新模型后，采用异步安全聚合算法对这些模型进行安全聚合(Secure Aggregation, SA)，形成当前训练轮次的全局模型。

SAFL-Server端算法(如算法1所示)：在每一轮训练中，动态选择客户端并行执行训练任务。当客户端完成训练任务后，它会将更新模型和训练轮次上传给服务端。服务端将异步接收到的客户端模型放入长度为K的缓冲队列Q中。一旦队列填满，服务端使用SA算法对缓冲队列中的模型进行安全聚合，并剔除恶意模型的影响。

2.1 客户端调度算法

SAFL-Client端算法(如算法2所示)：客户端接收到服务端下放的第t轮全局模型，并进行本地模型的迭代训练。本地每轮训练中，客户端根据预先设定的裁剪阈值C对模型梯度进行裁剪，得到裁剪

算法1 SAFL-Server算法

输入：数据集$\{D_1, D_2, \cdots, D_N\}$，学习率$\eta$，全局训练轮数$T$，缓冲区大小为$K$，客户端的数量为$N$，每轮选中的客户端占比为$F$。

输出：收敛后的全局模型ω_t

(1) **for** $t = 1$ to T **do**
(2) $k \leftarrow 0$
(3) $m \leftarrow \max(F \cdot N, 1)$
(4) $S_t \leftarrow \text{random}\{C_1, C_1, \cdots, C_m\} - \text{poisonerList}$
(5) **for** each client $i \in S_t$ in parallel **do**
(6) $\text{ClientUpdate}(i, w_t, t, \sigma_i)$
(7) **if** receive client update from client i **then**
(8) $(w_t^i, t_i) \leftarrow$received update from client i
(9) $k \leftarrow k + 1$
(10) **if** $k == K$ **then**
(11) $w_t^{\text{set}} \leftarrow \{w_t^1, w_t^2, \cdots, w_t^i\}$
(12) $\text{cid} \leftarrow \max(\cos(w_t^{\text{set}}, w_t))$ // 计算最大余弦相似度
(13) $w_{t+1} \leftarrow \text{SA}(\text{cid}, w_t^{\text{set}})$ // 安全聚合模型
(14) $k \leftarrow 0$
(15) **return** w_t

算法2　SAFL-Client

Thread-1 ClientUpdate:
输入：本地迭代轮次E，梯度裁剪阈值C，第t轮的全局模型w_t，本地差分隐私参数ε_i, δ_i。
输出：更新模型w_i和训练轮次t

(1) **for** $k=1$ to E **do**
(2)　**for** batch $b \subseteq D_i$ **do**
(3)　　$\boldsymbol{g}_{i,b}^{k}\left(\mathcal{D}_{i,b}\right) \leftarrow \nabla \ell\left(w, b\right)$ // 计算梯度
(4)　　$\boldsymbol{g}_{i,b}^{k}\left(\mathcal{D}_{i,b}\right) \leftarrow \boldsymbol{g}_{i,b}^{k}\left(\mathcal{D}_{i,b}\right) / \max\left(1, \frac{\|\boldsymbol{g}_{i,b}^{k}(\mathcal{D}_{i,b})\|_2}{C}\right)$ // 裁剪梯度
(5)　　$w_i^{k+1} \leftarrow w_i^k - \frac{1}{|D_i|}\sum_{b=1}^{|D_i|}\eta \boldsymbol{g}_{i,b}^{k}\left(D_{i,b}\right)$
(6)　　$w_i^{k+1} \leftarrow w_i^{k+1} + N\left(0, \Delta s^{D_i}/\varepsilon_i\right)$ // 添加高斯噪声
(7)　　**send** (w_i, t) to the server

Thread-2 ClientEvaluate
输入：诚实客户端cid，混淆模型集合garbleSet_t
输出：每个模型的预测准确率记录为列表scoreList_t
(8) local variables$l=0, \text{scoreList}_t=[\,]$
(9) **for** each model $m \in \text{garbleList}_t$ **do**
(10)　gcc $\leftarrow$ Evaluate (m, D_{cid}) // 计算模型对数据集D_{cid}的预测准确率
(11)　$\text{scoreList}_t[l] \leftarrow$ gcc
(12)　$l \leftarrow l+1$
(13) **return** scoreList_t

后的梯度$\boldsymbol{g}_i = \boldsymbol{g}_i / \max\left(1, \|\boldsymbol{g}_i\|_2 / C\right)$，使得梯度满足$\|\boldsymbol{g}_i\|_2 \le C$。为了保护用户的隐私，客户端在进行模型更新时采取了添加差分隐私噪声的措施。客户端根据设定的隐私预算ε_i和敏感度Δs生成高斯噪声，并将其添加到模型参数中。最后，客户端将添加噪声后的更新模型上传给服务端。

同时，被服务端选中的客户端还承担着检测投毒模型的任务。为了避免检测投毒任务对训练效率的影响，本文设计了一种加速训练的联邦学习系统。客户端并行运行两个进程：训练本地模型任务和计算模型预测准确率任务。在训练本地模型任务中，客户端根据接收到的全局模型进行本地模型的训练迭代，以提升模型的准确性和性能。在计算模型预测准确率任务中，客户端使用其他用户上传的模型对其本地数据进行预测，并获取预测准确率。客户端将这些预测准确率上传给服务端，以供服务端检测用户的投毒行为。

2.2　选择诚实客户端算法

在联邦学习中，攻击者上传恶意模型的目的是扰乱训练过程并使全局模型偏离其正常的更新方向[22]，最终导致模型不收敛。相反，诚实模型的均值聚合形成的全局模型与这些良性模型的更新方向更加一致。为了区分恶意模型和诚实模型的更新方向，采用余弦相似度算法来度量客户端模型与全局模型之间的相似度。余弦相似度计算两个向量之间的夹角，取值范围在[–1, 1]之间，值越接近1表示两个向量越相似。

在本文方案中，使用余弦相似度算法计算缓冲区队列Q中的各个客户端模型w_i与上一轮全局模型w_t之间的相似度。这种相似度度量了客户端模型与全局模型的更新方向的一致性程度。当相似度较高时，说明该客户端的本地模型与全局模型的更新方向更加一致，可以被认为是诚实客户端。为了选择目标诚实客户端，选择和全局模型相似度最大的客户端C_{cid}。这也意味着该客户端所拥有的本地数据更加符合整体数据的特征，可以被认为是良性数据。使用式(1)计算出目标诚实客户端的索引cid

$$\text{cid} = \max\left(\cos\left(w_t^{\text{set}}, w_t\right)\right) \tag{1}$$

其中，w_t是第t轮的全局模型，w_t^{set}表示缓冲队列Q中存放的客户端模型集合。

2.3　异步安全聚合算法

为了进一步检测出恶意用户，将利用筛选出具有良性数据的客户端对其他用户上传的模型进行评估。同时，将对缓冲队列Q中除了恶意模型以外的模型进行聚合，以生成下一轮的全局模型。

检测投毒用户(如算法3所示)：由于恶意模型的目的是破坏全局模型的收敛性，对良性数据的预测准确率往往较低。本文采用计算模型输出结果相似性的方法：根据选择诚实客户端算法确定出一个持有良性数据的客户端，使用K-Means算法将每个模型对该良性数据的预测准确率结果进行聚类，形成不同的簇；选择那些平均预测准确率较低的簇作为恶意模型集合poisonerSet_t。服务端将当前轮次收集到的训练模型mSet_t发送给诚实客户端。诚实客户端利用这些模型对本地的良性数据进行预测，并将预测结果返回给服务端。

本文采用混淆模型集合的方法来增强隐私保护性：随机生成一个符合正太分布的模型集合randomSet_t，

算法3　SAFL-Detect

输入：第t轮缓冲区中的模型集合 w_t^{set}
输出：投毒用户集合posionerSet_t

(1) $\text{randomModelSet}_t \leftarrow \{w_1, w_2, \cdots, w_i\}$
(2) $\text{garbelModelSet}_t \leftarrow \text{shuffle}(w_t^{\text{set}} + \text{randomModelSet}_t)$
(3) $\text{socreList}_t \leftarrow \text{ClientEvaluate}(\text{garbelModelSet}_t)$
(4) cluster1, cluster2 $\leftarrow$ K-Means(socreList_t)
(5) $\text{posionerSet}_t \leftarrow \min(\text{avg(cluster1)}, \text{avg(cluster2)})$
(6) **return** posionerSet_t

将其与集合mSet_t求并集，得到新的模型集合；服务端对模型集合进行随机打乱，得到混淆后的模型集合$\mathrm{garbleSet}_t$。即使是半诚实客户端接收到模型集合，也无法准确记录某个特定模型的信息。

聚合模型：本文采用加权聚合的方式来聚合出每一轮的全局模型。在异步联邦学习中，中央服务器通过指数衰减公式计算每一轮客户端上传的更新模型的陈旧性。对于客户端i，当它在第t_i轮接收到全局模型进行本地训练后，如果在同一轮上传本地模型，则陈旧程度$s=1$。相反，如果该客户端滞后了多轮，在第t轮上传了自己的模型，那么陈旧程度就为

$$s=(t-t_i)^{-\alpha} \tag{2}$$

其中，α是衰减系数，t为客户端上传模型时的全局轮次，t_i表示客户端下载训练模型时全局所在的轮次。

在聚合之前，根据检测投毒用户算法得到投毒用户集合$\mathrm{poisonerSet}_t$。然后对该集合进行遍历，更新攻击次数集合clientsAttack中用户的投毒次数。在聚合过程中，如果该用户被判定为恶意的次数超过阈值th，将不会参与到聚合任务中。同时，该用户会被彻底拉黑，后续不再参与训练。如算法4所示。

3 理论分析

本文采用本地差分隐私[23]的方法，在客户端的本地训练模型中添加满足(ε_i,δ_i)-本地差分隐私的高斯噪声，以保护用户模型隐私。在客户端训练模型阶段，客户端根据梯度裁剪阈值C对梯度进行裁剪，将其限制在阈值C以内，满足$\|\boldsymbol{g}_i\|_2\le C$。本地训练过程如式(3)所示

$$\begin{aligned} s^{D_i} &= w_i=\arg\min_w F(w,D_i)\\ &=\frac{1}{|D_i|}\sum_{j=1}^{|D_i|}\arg\min_w F_i(w,D_{i,j}) \end{aligned} \tag{3}$$

算法4 安全聚合算法SA

输入：第t轮接收到的客户端模型w_t^{set}，投毒次数集合clientsAttack。投毒次数的阈值th。客户端i的数据集样本数量$|D_i|$。t_i是客户端i接收模型时的全局轮次

输出：第t轮的全局模型w_t

(1) $\mathrm{poisonerSet}_t\leftarrow\mathrm{Detect}(w_t^{\mathrm{set}})$

(2) **for** each client id id $\in\mathrm{poisonerSet}_t$ **do**

(3) clientsAttack [id] $\leftarrow$ clientsAttack [id] $+1$

(4) **for** each client model $w_i\in w_t^{\mathrm{set}}$ **do**

(5) **if** clientsAttack $[i]<$ th **then**

(6) $s\leftarrow(1+t-t_i)^{-0.5}$

(7) $w_t\leftarrow w_{t-1}+s\cdot(|D_i|/M)\,w_i$

(8) **return** w_t

其中，模型训练数据集为D_i，$D_{i,j}$是D_i中第j个样本，F_i是该模型的损失函数。

根据差分隐私的定义，对于一对相邻数据集D_i和D_i'，它们之间仅有一处数据不同。在本地的每轮训练中，可以计算客户端C_i的敏感度Δs^{D_i}为

$$\begin{aligned} \Delta s^{D_i} &= \max_{D_i,D_i'}\left\|\Delta s_U^{D_i}-\Delta s_U^{D_i'}\right\|\\ &=\max\left\|\frac{1}{|D_i|}\sum_{j=1}^{|D_i|}\arg\min_w F_i(w,D_{i,j})\right.\\ &\quad\left.-\frac{1}{|D_i'|}\sum_{j=1}^{|D_i'|}\arg\min_w F_i\left(w,D_{i,j}'\right)\right\|\\ &=\frac{2C}{|D_i|} \end{aligned} \tag{4}$$

根据式(4)，可以得出客户端每轮训练时的敏感度$\Delta s=\max\left\{\Delta s^{D_i}\right\}$。最后，使用敏感度$\Delta s$计算出本轮训练中要添加的高斯噪声的标准差，如式(5)所示

$$\sigma_i=\frac{c\cdot\Delta s}{\varepsilon_i},\Delta s=\frac{2C}{|D_i|},c\ge\sqrt{2\ln(1.25/\delta)} \tag{5}$$

在服务端聚合模型阶段，本文采用的聚合模型算法为

$$\begin{aligned} w_t&=\frac{s_1|D_1|}{M}w_{11}+\frac{s_2|D_2|}{M}w_2+\cdots+\frac{s_K|D_K|}{M}w_K\\ &=\frac{s_i|D_1|}{M}(w_1+w_2+\cdots+w_k) \end{aligned} \tag{6}$$

通过聚合公式，可以得出聚合模型的查询敏感度$\Delta s_G^{D_i}$为

$$\begin{aligned} \Delta s_G^{D_i} &= \max_{D_i,D_i'}\|\Delta s^{D_i}-\Delta s^{D_i'}\|\\ &=\max(s_i)\frac{2C}{|D_i|}=\frac{s\cdot 2C}{|D_i|} \end{aligned} \tag{7}$$

可以得出查询聚合模型的最大敏感度$\Delta s_G=\max\left\{\Delta s_G^{D_i}\right\}$。

为了保证每一轮全局模型聚合后的噪声满足本地差分隐私，需要确保客户端在本地添加的高斯噪声标准差满足$\sigma\ge c\Delta s_G/\varepsilon=2cCs/\varepsilon|D_i|$和$c\ge\sqrt{2\ln(1.25/\delta)}$。由差分隐私的串行组合性质可知，在服务端全局经过$T$轮的迭代后，最终的噪声满足$(T\varepsilon,T\delta)$-本地差分隐私。以下附上简要证明。

证明

$$\begin{aligned} c_M(o,D,D') &= \ln\frac{\Pr[\mathcal{M}(D)=o]}{\Pr[\mathcal{M}(D')=o]}=\left|\ln\frac{\mathrm{e}^{-\frac{(x)^2}{2\sigma^2}}}{\mathrm{e}^{-\frac{(x+\Delta s_G)^2}{2\sigma^2}}}\right|\\ &=\left|\frac{1}{2\sigma^2}\left(2x\Delta s_G+(\Delta s_G)^2\right)\right|\le\varepsilon \end{aligned} \tag{8}$$

因为$x < \sigma^2\varepsilon/\Delta s_G - \Delta s_G/2$，为了保证隐私损失被限制为$\varepsilon$的概率至少为$1-\delta$，所以有

$$\Pr\left[|x| \geq \frac{\sigma^2\varepsilon}{\Delta s_G} - \frac{\Delta s_G}{2}\right] < \delta$$
$$\Rightarrow \Pr\left[x \geq \frac{\sigma^2\varepsilon}{\Delta s_G} - \frac{\Delta s_G}{2}\right] < \frac{\delta}{2} \tag{9}$$

令$t = \left(\sigma^2\varepsilon/\Delta s_G\right) - \left(\Delta s_G/2\right)$，根据高斯分布的概率密度函数有

$$\Pr[x \geq t] = \int_t^{\infty} \frac{1}{\sigma\sqrt{2\pi}} \mathrm{e}^{-\frac{(x-\mu)^2}{2\sigma^2}} \mathrm{d}x \leq \frac{\sigma}{t\sqrt{2\pi}} \mathrm{e}^{-\frac{t^2}{2\sigma^2}} \tag{10}$$

所以有

$$\frac{\sigma}{t\sqrt{2\pi}} \mathrm{e}^{-\frac{t^2}{2\sigma^2}} < \frac{\delta}{2} \Leftrightarrow \frac{\sigma}{t} \mathrm{e}^{-\frac{t^2}{2\sigma^2}} < \frac{\delta\sqrt{2\pi}}{2}$$
$$\Leftrightarrow \ln\left(\frac{t}{\sigma}\right) + \frac{t^2}{2\sigma^2} > \ln\left(\frac{2}{\delta\sqrt{2\pi}}\right) \tag{11}$$

将$\sigma = c\Delta s/\varepsilon$带入到上式中，将会得到$c \geq \sqrt{2\ln(1.25/\delta)}$，这满足高斯机制的差分隐私条件。

4　实验

4.1　实验设置

本文中所有实验都采用了交叉熵损失函数和随机梯度下降(Stochastic Gradient Descent, SGD)方法。测试数据集被随机打乱并分成N份，然后将这些数据分发给N个客户端，用于本地模型训练。训练数据集分别采用MNIST和Fashion-MNIST，标签类别均为10，学习率分别为0.01和0.005。实验共计模拟了100个客户端，并在每次训练中随机选择一部分比例为F的客户端参与训练，将训练结果返回服务端进行模型聚合。在评估异步联邦学习下模型的陈旧程度时，衰减系数通常在范围$\alpha \in (0,1)$内选择。根据Xie等人[4]的实验研究，采用衰减系数$\alpha = 0.5$可提高训练后的模型预测准确率。因此，在实验中将采取衰减算法$s = (1+t-t_i)^{-0.5}$。

4.2　抵御投毒攻击

本方案设定攻击者比例为p，数值为0.1, 0.2, 0.3, 0.4, 0.5和0.6，并选择卷积神经网络(Convolutional Neural Network, CNN)和多层感知机(MultiLayer Perceptron, MLP)这作为异步联邦学习的训练模型，并对针对手写数字数据集(Modified National Institute of Standards and Technology database, MNIST)和MNIST替代品(Fashion-Modified National Institute of Standards and Technology database, Fashion-MNIST)这两种数据集进行训练学习。

在针对投毒攻击的防御方面，实验中采取了一种动态拉黑策略，即在训练开始时将所有参与训练的客户端投毒次数初始化为0。如果某用户在$\mathrm{poisonerSet}_t$中的次数超过阈值th的话，该用户将会被拉黑，并在后续训练中不再参与。实验结果如图2所示，当阈值设置为$\mathrm{th}=2$时，全局模型最早收敛，且预测准确率最高。此外，该设置还能减小预测准确率的波动，表现出更好的实验效果。

在实验中，针对CNN模型，设定全局训练轮次为50次，表示在服务端进行了50轮的迭代聚合。针对MLP模型，注意到较少的训练轮次可能导致全局模型未能充分收敛的情况。为了达到模型收敛的效果，将MLP模型的全局训练轮次设定为100次。

根据实验结果(如图3所示)，随着投毒攻击比例p的增加，可能会观察到训练初期的预测准确率波动。这是因为恶意模型对全局模型的训练造成了一定干扰。然而，通过本方案的安全聚合算法，随着训练的进行，该方案逐渐识别出所有的投毒攻击者并将其排除在训练之外。全局模型逐渐趋于稳定，并且预测准确率逐步提升并保持稳定状态。本方案有效地减轻了投毒攻击对全局模型训练的干扰，使得训练模型能够在更短的时间内达到较高的性能水平，并且保持相对稳定的预测能力。

针对MNIST数据集的实验结果表明，在不同的投毒比例p下，CNN模型的最终预测准确率都稳定在95%左右。这说明CNN模型在面对投毒攻击时具有较好的鲁棒性。相比之下，MLP模型在初期预测准确率较低。随着训练的不断迭代，SA安全聚合算法识别出所有恶意用户。MLP模型也逐步收敛并达到较高的预测准确率。在使用Fashion-MNIST数据集进行训练时，CNN模型比MLP模型展现出更好的学习能力，表现为更快的收敛速度和更高的预测准确率。

通过与现有方案联邦平均聚合(Federated Averaging, FedAvg)[24]以及安全聚合方案Krum[25], Trimmed-mean[26]进行对比实验，来评估本方案在抵御投毒攻击方面的效果。每轮随机抽取20位客户

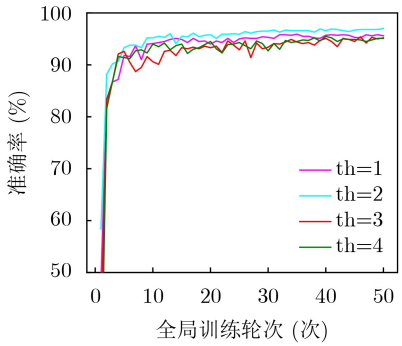


图 2 不同投毒阈值下的预测准确率对比

端参与训练，并设置缓冲区Q存放最大模型个数K为10，以接收所有客户端异步发送过来的最新模型。

图4展示了3种方案在不同数据集和训练模型下的预测准确率曲线。本文方案能够有效地剔除恶意用户，保证了全局模型的稳定和最快的收敛性。FedAvg方案[24]采用的平均加权方法会将恶意模型与诚实模型进行聚合，存在预测准确率抖动起伏大、模型收敛效果不佳的问题。Trimmed-mean方案[26]将各个客户端模型每个维度的中值作为对应的聚合模型参数，存在较多恶意模型会表现为聚合模型的收敛速度较慢。Krum方案[25]在初期有较高的预测准确率，在恶意模型较多时，Krum会将某个恶意模型作为最相似模型，并将其作为全局模型下放给客户端。

表2总结了在MNIST和Fashion-MNIST数据集下，投毒比例为0.4的情况下，各个聚合方案的预测准确率。SAFL方案表现出最快的收敛速度和最高的预测准确率。使用CNN模型时，SAFL方案的预测准确率高达96.5%，相较于在 FedAvg方案提升了约19%。此外，与Krum和Trimmed-mean方案相比，SAFL方案分别提升了约846%和85.9%。在使用MLP模型进行训练的场景下，SAFL方案的预测准确率相对于FedAvg提升了约20%，同时比Krum和Trimmed-mean分别提升了214%和21.6%。在提升模型预测准确率方面，SAFL方案相较于FedAvg，在CNN模型和MLP模型下分别提升了92.7%和14.2%。与Krum方案相比，SAFL方案在CNN模型和MLP模型下分别提升了739%和148%。与Trimmed-mean方案对比，SAFL方案在CNN模型和MLP模型下分别提升了104.7%和14.5%。

5 结束语

异步联邦学习作为加速联邦学习的新兴方案，受到很多研究学者的关注。本文针对投毒攻击以及参与方可能从全局模型中提取敏感信息的隐私泄露问题，提出了一个抗投毒攻击的异步联邦学习方案SAFL，有效降低了投毒攻击者对全局模型的干扰，增强全局模型的鲁棒性。同时，引入了差分隐私技术，通过对参与方本地的训练模型添加本地差分隐私噪声，来隐藏敏感信息实现对隐私的保护。未来的研究将重点放在非独立同分布的数据集上进行训练，以更好地适应这些数据集的特点，考虑结合个性化模型聚合策略、动态选择参与者等技术，提高异步联邦学习在非独立同分布数据上抵御投毒攻击的性能和训练效果。

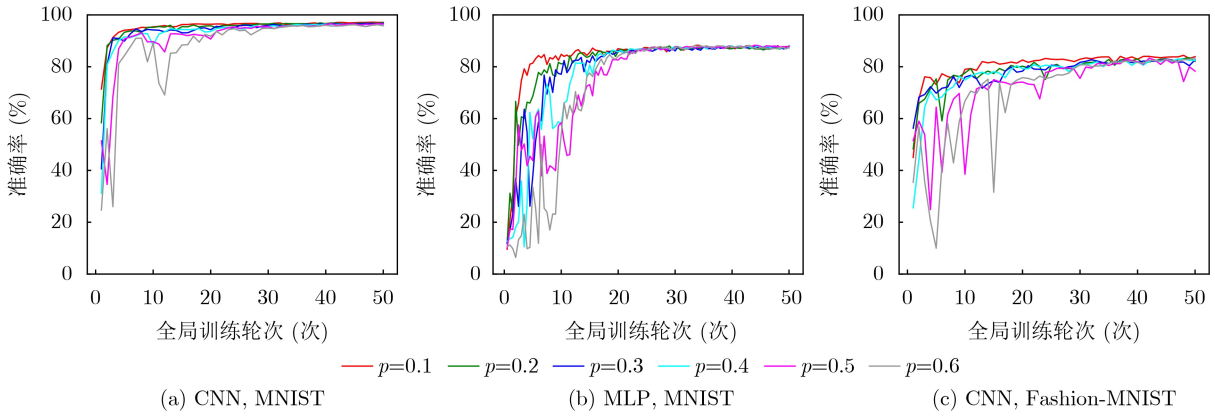


图 3 不同测试数据集和训练模型下抵御投毒攻击效果对比，全局隐私预算$\varepsilon = 6$

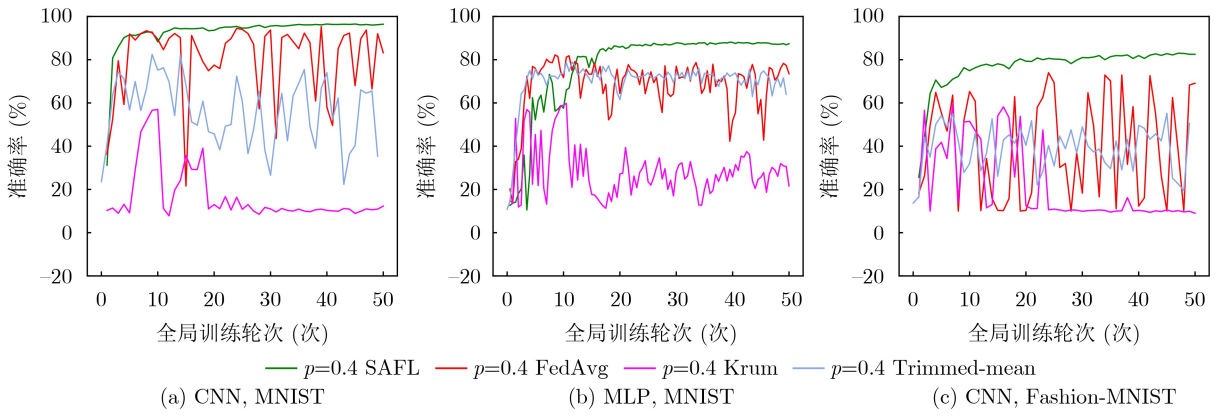


图 4 对比SAFL和FedAvg,Krum以及Trimmed-mean方案在不同数据集下的抗投毒预测，全局差分隐私预算$\varepsilon = 6$

表 2　不同数据集上模型的预测准确率，投毒比例p=0.4 (%)

训练模型	方法	MNIST 下准确率	Fashion-MNIST 下准确率
CNN	FedAvg[24]	81.1	42.7
	Krum[25]	10.2	9.8
	Trimmed-mean[26]	51.9	40.2
	SAFL	**96.5**	**82.3**
MLP	FedAvg[24]	72.3	66.7
	Krum[25]	27.5	30.7
	Trimmed-mean[26]	71.0	66.5
	SAFL	**86.4**	**76.2**

参 考 文 献

[1] MCMAHAN B, MOORE E, RAMAGE D, *et al*. Communication-efficient learning of deep networks from decentralized data[C]. The 20th International Conference on Artificial Intelligence and Statistics, Fort Lauderdale, USA, 2017: 1273–82.

[2] YANG Qiang, LIU Yang, CHEN Tianjian, *et al*. Federated machine learning: Concept and applications[J]. *ACM Transactions on Intelligent Systems and Technology*, 2019, 10(2): 12. doi: 10.1145/3298981.

[3] BONAWITZ K, EICHNER H, GRIESKAMP W, *et al*. Towards federated learning at scale: System design[C]. Machine Learning and Systems, Stanford, USA, 2019: 374–88.

[4] XIE Cong, KOYEJO S, and GUPTA I. Asynchronous federated optimization[EB/OL]. https://arxiv.org/abs/1903.03934, 2019.

[5] LIU Jianchun, XU Hongli, WANG Lun, *et al*. Adaptive asynchronous federated learning in resource-constrained edge computing[J]. *IEEE Transactions on Mobile Computing*, 2023, 22(2): 674–690. doi: 10.1109/TMC.2021.3096846.

[6] HUBA D, NGUYEN J, MALIK K, *et al*. Papaya: Practical, private, and scalable federated learning[C]. Machine Learning and Systems, Santa Clara, USA, 2022: 814–32.

[7] BAGDASARYAN E, VEIT A, HUA Yiqing, *et al*. How to backdoor federated learning[C]. The Twenty Third International Conference on Artificial Intelligence and Statistics, Palermo, Italy, 2020: 2938–2948.

[8] 高莹, 陈晓峰, 张一余, 等. 联邦学习系统攻击与防御技术研究综述[J]. 计算机学报, 2023, 46(9): 1781–1805. doi: 10.11897/SP.J.1016.2023.01781.
GAO Ying, CHEN Xiaofeng, ZHANG Yiyu, *et al*. A survey of attack and defense techniques for federated learning systems[J]. *Chinese Journal of Computers*, 2023, 46(9): 1781–1805. doi: 10.11897/SP.J.1016.2023.01781.

[9] 汤凌韬, 陈左宁, 张鲁飞, 等. 联邦学习中的隐私问题研究进展[J]. 软件学报, 2023, 34(1): 197–229. doi: 10.13328/j.cnki.jos.006411.
TANG Lingtao, CHEN Zuoning, ZHANG Lufei, *et al*. Research progress of privacy issues in federated learning[J]. *Journal of Software*, 2023, 34(1): 197–229. doi: 10.13328/j.cnki.jos.006411.

[10] SO J, NOLET C J, YANG C S, *et al*. Lightsecagg: A lightweight and versatile design for secure aggregation in federated learning[C]. Machine Learning and Systems, Santa Clara, USA, 2022: 694–720.

[11] FANG Minghong, LIU Jia, GONG N Z, *et al*. AFLGuard: Byzantine-robust asynchronous federated learning[C]. The 38th Annual Computer Security Applications Conference, Austin, USA, 2022: 632–646. doi: 10.1145/3564625.3567991.

[12] WANG Rong and TSAI W T. Asynchronous federated learning system based on permissioned blockchains[J]. *Sensors*, 2022, 22(4): 1672. doi: 10.3390/s22041672.

[13] LU Yunlong, HUANG Xiaohong, DAI Yueyue, *et al*. Differentially private asynchronous federated learning for mobile edge computing in urban informatics[J]. *IEEE Transactions on Industrial Informatics*, 2020, 16(3): 2134–2143. doi: 10.1109/TII.2019.2942179.

[14] DAMASKINOS G, EL MHAMDI E M, GUERRAOUI R, *et al*. Asynchronous Byzantine machine learning (the case of SGD)[C]. The 35th International Conference on Machine Learning, Stockholmsmässan, Sweden, 2018: 1153–1162.

[15] 刘艺璇, 陈红, 刘宇涵, 等. 联邦学习中的隐私保护技术[J]. 软件学报, 2022, 33(3): 1057–1092. doi: 10.13328/j.cnki.jos.006446.
LIU Yixuan, CHEN Hong, LIU Yuhan, *et al*. Privacy-preserving techniques in federated learning[J]. *Journal of Software*, 2022, 33(3): 1057–1092. doi: 10.13328/j.cnki.jos.006446.

[16] WANG Bo, LI Hongtao, GUO Yina, *et al*. PPFLHE: A privacy-preserving federated learning scheme with homomorphic encryption for healthcare data[J]. *Applied Soft Computing*, 2023, 146: 110677. doi: 10.1016/j.asoc.2023.110677.

[17] FENG Jun, YANG L T, ZHU Qing, *et al*. Privacy-preserving tensor decomposition over encrypted data in a federated cloud environment[J]. *IEEE Transactions on Dependable and Secure Computing*, 2020, 17(4): 857–868. doi: 10.1109/TDSC.2018.2881452.

[18] 李腾, 方保坤, 马卓, 等. 基于同态加密的医疗数据密文异常检测方法[J]. 中国科学:信息科学, 2023, 53(7): 1368–1391. doi: 10.1360/ssi-2022-0214.
LI Teng, FANG Baokun, MA Zhuo, *et al*. Homomorphic encryption-based ciphertext anomaly detection method for

e-health records[J]. *Scientia Sinica (Informationis)*, 2023, 53(7): 1368–1391. doi: 10.1360/ssi-2022-0214.

[19] GEHLHAR T, MARX F, SCHNEIDER T, *et al*. SafeFL: MPC-friendly framework for private and robust federated learning[C]. 2023 IEEE Security and Privacy Workshops (SPW), San Francisco, USA, 2023: 69–76. doi: 10.1109/SPW59333.2023.00012.

[20] MANSOURI M, ÖNEN M, JABALLAH W B, *et al*. Sok: Secure aggregation based on cryptographic schemes for federated learning[J]. *Proceedings on Privacy Enhancing Technologies*, 2023, 2023(1): 140–157. doi: 10.56553/popets-2023-0009. doi: 10.56553/popets-2023-0009. doi: 10.56553/popets-2023-0009.doi:10.56553/popets-2023-0009.

[21] FENG Jun, YANG L T, NIE Xin, *et al*. Edge–cloud-aided differentially private tucker decomposition for cyber–physical–social systems[J]. *IEEE Internet of Things Journal*, 2022, 9(11): 8387–8396. doi: 10.1109/JIOT.2020.3004826.

[22] CAO Di, CHANG Shan, LIN Zhijian, *et al*. Understanding distributed poisoning attack in federated learning[C]. IEEE 25th International Conference on Parallel and Distributed Systems (ICPADS), Tianjin, China, 2019: 233–239. doi: 10.1109/ICPADS47876.2019.00042.

[23] ABADI M, CHU A, GOODFELLOW I, *et al*. Deep learning with differential privacy[C]. The 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, 2016: 308–318. doi: 10.1145/2976749.2978318.

[24] KONEČNÝ J, MCMAHAN H B, XU F X, *et al*. Federated learning: Strategies for improving communication efficiency[C]. 6th International Conference on Learning Representations, Vancouver, BC, Canada, 2016.

[25] BLANCHARD P, EL MHAMDI E M, GUERRAOUI R, *et al*. Machine learning with adversaries: Byzantine tolerant gradient descent[C]. The 31st International Conference on Neural Information Processing Systems, Long Beach, USA, 2017: 118–128.

[26] YIN Dong, CHEN Yudong, KANNAN R, *et al*. Byzantine-robust distributed learning: Towards optimal statistical rates[C]. The 35th International Conference on Machine Learning, Stockholmsmässan, Sweden, 2018: 5650–5659.

魏立斐：男，博士，教授，研究方向为信息安全、隐私保护、密码学.
张无忌：男，硕士生，研究方向为信息安全、联邦学习.
张 蕾：女，博士，副教授，研究方向为密码学、数据安全、访问控制、联邦学习.
胡雪晖：女，博士，研究方向为隐私保护、数据安全.
王绪安：男，博士，教授，研究方向为信息安全、密码学.

责任编辑：余 蓉