计算机科学
COMPUTER SCIENCE

PRFL:一种隐私保护联邦学习鲁棒聚合方法

高琦, 孙奕, 盖新貌, 王友贺, 杨帆

引用本文

高琦, 孙奕, 盖新貌, 王友贺, 杨帆. PRFL:一种隐私保护联邦学习鲁棒聚合方法[J]. 计算机科学, 2024, 51(11): 356-367.

GAO Qi, SUN Yi, GAI Xinmao, WANG Youhe, YANG Fan. PRFL:Privacy-preserving Robust Aggregation Method for Federated Learning [J]. Computer Science, 2024, 51(11): 356-367.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.231000158

PRFL:一种隐私保护联邦学习鲁棒聚合方法

高 琦[1] 孙 奕[1] 盖新貌[3] 王友贺[1] 杨 帆[1,2]
1 信息工程大学密码工程学院 郑州 450001
2 中国人民解放军 61623 部队 北京 100036
3 中国人民解放军 93216 部队 北京 100085
(qig_57@163.com)

摘 要 联邦学习允许用户通过交换模型参数共同训练一个模型,能够降低数据泄露风险。但研究发现,通过模型参数仍能推断出用户隐私信息。对此,许多研究提出了模型隐私保护聚合方法。此外,恶意用户可通过提交精心构造的投毒模型破坏联邦学习聚合,且模型在隐私保护下聚合,恶意用户可以实施更加隐蔽的投毒攻击。为了在实现隐私保护的同时抵抗投毒攻击,提出了一种隐私保护联邦学习鲁棒聚合方法 PRFL。PRFL 不仅能够有效防御拜占庭用户发起的投毒攻击,还保证了本地模型的隐私性、全局模型的准确性和高效性。首先,提出了一种双服务器结构下轻量级模型隐私保护聚合方法,实现模型隐私保护聚合,同时保证全局模型的准确性并且不会引入开销问题;然后,提出了一种密态模型距离计算方法,在不暴露本地模型参数的同时允许双方服务器计算出模型距离,并基于该方法和局部离群因子算法(Local Outlier Factor,LOF)设计了一种投毒模型检测方法;最后,对 PRFL 的安全性进行了分析。在两种真实图像数据集上的实验结果表明:无攻击时,PRFL 可以取得与 FedAvg 相近的模型准确率;PRFL 在数据独立同分布(IID)和非独立同分布(Non-IID)设置下能有效防御 3 种先进的投毒攻击,并优于现有的 Krum,Median,Trimmed mean 方法。

关键词:联邦学习;隐私保护;投毒攻击;鲁棒聚合;离群值
中图分类号 TP391

PRFL:Privacy-preserving Robust Aggregation Method for Federated Learning

GAO Qi[1],SUN Yi[1],GAI Xinmao[3],WANG Youhe[1] and YANG Fan[1,2]
1 School of Cryptography Engineering,Information Engineering University,Zhengzhou 450001,China
2 Unit 61623,Beijing 100036,China
3 Unit 93216,Beijing 100085,China

Abstract Federated learning allows users to train a model together by exchanging model parameters and can reduce the risk of data leakage. However,studies have found that user privacy information can still be inferred through model parameters,and many studies have proposed model privacy-preserving aggregation methods. Moreover,malicious users can corrupt federated learning aggregation by submitting carefully constructed poisoning models,and with models aggregated under privacy protection,malicious users can implement more hidden poisoning attacks. In order to implement privacy protection while resisting poisoning attacks,a privacy-preserving federated learning robust aggregation method named PRFL is proposed. PRFL can not only effectively defends against poisoning attacks launched by Byzantine users,but also guarantee the privacy of the local model,the accuracy and efficiency of the global model. Specifically,a lightweight model privacy-preserving aggregation method under dual-server architecture is first proposed to achieve the privacy-preserving aggregation of the model,while guaranteeing the accuracy of global model without introducing overhead problems. Then a secret model distance computation method is proposed,which allows both servers to compute model distances without exposing the local model parameters,and poisoning model detection method is designed based on this method and local outlier factor(LOF) algorithm. Finally,security of PRFL is analysed. Experimental results on two real image datasets show that PRFL can obtain similar model accuracy to FedAvg under no attack,and PRFL can effectively defend against three advanced poisoning attacks and outperform existing Krum,Median,and Trimmed mean methods in both the data independent identically distributed(IID) and non-IID settings.

Keywords Federated learning,Privacy protection,Poisoning attack,Robust aggregation,Outlier

到稿日期:2023-10-23 返修日期:2024-03-29
通信作者:孙奕(11112072@bjtu.edu.cn)

1　引言

联邦学习[1]是谷歌公司提出的一种分布式机器学习训练框架。在不暴露本地数据的情况下，它允许用户通过交换模型的方式共同训练一个全局模型。在联邦学习中，训练过程由中心服务器负责协调。用户首先利用本地数据和从中心服务器收到的全局模型来训练本地模型，然后将本地模型提交给中心服务器，最后由中心服务器聚合出全局模型并再次下发给用户。随着与数据隐私相关的法律条例相继出台[2]，数据交换范围被严格限制，联邦学习在各个领域被广泛研究和应用，例如单词预测、医疗保健、智能交通等。但研究发现，联邦学习很容易受到隐私和投毒攻击[3-4]。

在隐私方面，一些研究[5-7]发现，通过对本地模型进行分析可以推断出用户本地训练数据的敏感信息，甚至可以重构出原始数据[8]。为了解决隐私问题，模型隐私保护聚合技术被广泛研究[9]，其主要目标是实现在服务器不知道单个本地模型的情况下完成模型聚合。这些解决方案一般包含差分隐私[10-11]、安全多方计算[12]以及同态加密[13-15]等技术。差分隐私是通过添加随机噪声来保护模型隐私，通过数学形式化地描述了隐私保护水平，具有高效性，但噪声的引入会损害全局模型质量。安全多方计算和同态加密通过密码学等技术实现模型隐私保护聚合，并且基本不损害模型质量。但是由于它们依赖复杂的密码原语，因此会给用户引入大量的计算开销和通信开销。

除了隐私问题，联邦学习也容易遭受投毒攻击。由于服务器无法控制用户的行为，拜占庭用户可能偏离规定的训练规则，提交虚假或者恶意的本地模型来破坏联邦学习，导致全局模型偏离正确的收敛方向。为了防御投毒攻击，一些研究基于相似性[16-17]、统计值[18]、梯度裁剪[19]等方式来检测投毒模型或者缓解投毒模型带来的影响，增强聚合的鲁棒性。然而，这些检测和防御方法需要服务器能够以明文的方式访问本地模型参数，这违背了模型隐私保护要求。上述方法在解决聚合鲁棒性问题时可能会带来隐私问题。而在隐私保护聚合中，由于本地模型被保护，因此拜占庭用户可以更隐蔽地实施投毒攻击。

同时保证隐私性和鲁棒性，是联邦学习面临的严峻挑战。一些研究[20-24]尝试同时解决这两个问题。其中，文献[21-22]利用同态加密和安全多方计算技术允许聚合服务器通过梯度密文计算模型相似度，在保证梯度隐私的同时降低或者消除投毒模型带来的影响。但这要求用户对梯度进行加密计算，对于资源受限的设备来说可能是无法接受的。文献[20,24]中每轮需要用户之间进行多次交互，从而协助聚合服务器识别投毒模型，同时保证本地模型不会被泄露。文献[23]证明了差分隐私技术可以同时提高隐私性和鲁棒性，但过多的噪声可能会严重降低全局模型的准确性。

为了能够同时保证隐私性和鲁棒性，并且不损害全局模型的准确性以及引入过多的计算和通信开销，本文提出了一种隐私保护联邦学习鲁棒聚合方法 PRFL，它能够实现模型隐私保护聚合，同时有效防御拜占庭用户的投毒攻击，支持数据 IID 和 Non-IID 的设置。无攻击时，PRFL 基本不会降低模型的准确性，不会给用户引入不可接受的计算开销和通信开销。具体来说，设计了一种双服务器结构下具有投毒模型检测的轻量级模型隐私保护聚合方法。首先，通过模型划分和轻量级加密实现模型隐私保护；然后，提出了一种密态模型距离计算方法，在不暴露模型参数的同时，允许双方服务器计算模型之间的距离；最后，基于该方法和 LOF 算法[25]，设计了一种投毒攻击检测方法。

综上所述，本文的主要贡献如下：

1）提出了一种双服务器结构下具有投毒模型检测功能的轻量级模型隐私保护聚合方法。该方法可在实现本地模型隐私保护聚合的同时有效检测投毒模型，保证模型的准确性且不给用户引入开销问题，可以容忍用户随时退出。

2）提出了一种密态模型距离计算方法，其在不暴露本地模型参数的同时允许双方服务器计算出模型距离；并基于该方法和 LOF 算法设计了一种投毒模型检测方法，该方法可有效防御数据 IID 和 Non-IID 设置下拜占庭用户发起的投毒攻击。

3）对方案的安全性进行了分析，在两种真实图像数据集上对方案的鲁棒性和准确性进行了评估。实验结果表明，本文方案具有可行性和鲁棒性，无攻击时，可以取得与 FedAvg 相近的模型准确率，对 3 种投毒攻击具有鲁棒性，优于 Krum，Median 和 Trimmed mean 方法。

2　相关研究

联邦学习作为一种隐私保护机器学习范式被广泛研究和应用，数据不出本地的方式大大降低了数据泄露的风险。然而，已有研究发现通过模型的梯度或者参数仍可以推理出用户本地的训练样本信息。一些研究[12,26-27]被提出用于解决联邦学习中模型泄露隐私的问题。

除了隐私性，联邦学习也容易遭受投毒攻击[28-32]。一些研究[16-19]提出了拜占庭鲁棒聚合方法来防御投毒攻击。但在模型隐私保护聚合中，由于服务器无法获取到本地模型参数，这些检测方法都无法使用。

为了能够同时保证隐私性和鲁棒性，近年来学者们进行了一些相关研究[20-24,33-34]。文献[23]讨论了通过差分隐私来同时解决隐私性和鲁棒性问题，实验结果表明了差分隐私技术的可行性，但会明显降低全局模型的质量。文献[33]结合差分隐私洗牌模型和 RSA 聚合机制[29]来保护模型隐私并抵抗拜占庭攻击，但是引入的噪声仍会对全局模型的准确性造成影响。因此，差分隐私虽然一定程度上可以降低隐私泄露风险并抵抗投毒攻击，且不会给用户带来开销问题，但需要考虑模型性能和噪声水平之间的平衡。

文献[20]提出了一种基于 Shamir 秘密分享的拜占庭安全聚合框架 BREA。首先，用户通过对模型参数进行秘密分享实现模型隐私保护；然后，服务器利用 Shamir 秘密分享的同态性计算模型之间的距离，整个聚合过程需要服务器和用户之间进行多次交互。此外，该方案需要用户对每个模型参数进行秘密分享。文献[22]提出了 PEFL 来解决隐私性和鲁棒性问题。PEFL 采用线性同态加密保护模型隐私，然后利用两个非共谋服务器相互协作，通过 4 个交互协议计算模型

参数之间的皮尔逊相关系数来过滤投毒模型。ShieldFL[21]同样基于两个非共谋服务器,然后采用基于 Paillier 的双陷门同态方案来实现安全和隐私保护的联邦学习。ShieldFL 基于梯度密文计算本地更新的余弦相似度,然后通过余弦相似度计算本地更新权重来降低投毒模型的影响。文献[34]提出了一种分层隐私保护防御机制 APFed,APFed 底层采用 Paillier 同态加密保护模型隐私,并通过 BatchCrypt 来降低通信开销和计算开销。通过将边缘节点聚类和分层,在不暴露模型参数的情况下计算出每位参数的中位数,并以此为基准计算本地模型的余弦相似度,实现了在数据 IID 和 Non-IID 下的鲁棒聚合。上述方案[20-22,34]都是基于密码学和安全多方计算技术对模型参数进行保护,并在密文下检测投毒模型,可以保证模型隐私性和聚合的鲁棒性,同时不损害全局模型性能。但是它们需要用户对梯度进行复杂的加密操作,这可能会给用户带来无法接受的计算开销和通信开销。文献[24]则提出了一种新颖的联邦学习模式 Fragmented Federated Learning (FFL)。其中,用户在上传梯度之前通过轻量级两方计算协议基于信誉选择用户交换模型片段,混淆本地模型参数,解除模型参数和用户之间的关系,从而防止服务器获取到完整的本地模型;服务器基于混淆模型计算模型余弦相似度,从而检测投毒模型。但前期需要用户之间进行协商配对,这可能会给用户带来通信开销。

上述研究虽然能够保证模型的隐私性和聚合的鲁棒性,但可能会带来新的问题,例如引入过多的计算开销和通信开销,损坏模型性能等。为了能够在保证隐私性和鲁棒性的同时避免用户开销增大和全局模型准确率显著降低的问题,本文提出了一种隐私保护联邦学习鲁棒聚合方法。用户以较低的计算开销和通信开销确保在不暴露本地模型参数的情况下,服务器能够实现对投毒模型的检测以及模型的隐私保护聚合,同时不会显著降低全局模型的准确性。

3 背景知识

3.1 投毒攻击

由于服务器无法控制用户的行为,因此拜占庭用户可能会偏离规定的训练协议,通过提交恶意的本地模型来攻击全局模型。根据攻击目的,投毒攻击可分无目标攻击[28-30]和目标攻击[31-32]。其中,无目标攻击主要是针对模型可用性的攻击,例如阻止模型收敛或者聚合出无效的全局模型;目标攻击则是使全局模型在预测时偏向某个类别,使其对一些特定输入做出错误的预测。

根据攻击方式,投毒攻击可以分为模型投毒[29]和数据投毒[35-36]。在模型投毒中,攻击者在向服务器发送模型之前会修改模型参数,例如高斯攻击、符号翻转攻击;在数据投毒中,攻击者在训练其本地模型之前,会改变训练集中数据的某些属性,例如标签翻转攻击中修改某些类别数据的标签。

3.2 LOF 算法

LOF[25]是一种基于密度的离群点检测算法,它通过数据样本及其周围点的密度关系计算数据点的离群因子来反映数据样本的异常程度。离群因子的计算是基于样本之间的距离,因此,在计算离群因子之前需要先计算出样本之间的距离。这个距离可以是欧氏距离、汉明距离以及马氏距离等。假设样本集合为 O,样本之间的距离表示为 $d_{i,j}$,在计算出样本距离的基础上,离群因子计算步骤如下。

1)计算第 k 距离 d_k^i:按照升序对样本之间的距离 $d_{i,j}$ $(i,j\in U)$ 进行排序,选择第 k 个距离值作为样本点 i 的 d_k^i,则点 i 的第 k 邻域为:

$$N_k^i=\{j\},j\in O,d_{i,j}\leqslant d_k^i \tag{1}$$

2)计算第 k 个可达距离 $rd_k(i,j)$:样本点 j 到点 i 的 $rd_k(i,j)$ 是点 j 的第 k 距离和点 i 和 j 之间真实距离 $d_{i,j}$ 的较大值。

$$rd_k(i,j)=\max\{d_{i,j},d_k^i\},i,j\in O,i\neq j \tag{2}$$

3)计算局部可达密度 lrd:样本点 i 的 lrd 是点 i 第 k 邻域 N_k^i 内所有样本点到样本点 i 的平均第 k 可达距离的倒数。

$$lrd_i=\frac{N_k^i}{\sum\limits_{j\in N_k^i}rd_k(j,i)}. \tag{3}$$

可以看出,lrd_i 代表了点 i 的密度,点 i 与其他样本点距离越近,$rd_k(j,i)$ 越可能是较小第 k 距离,即 lrd 值越大,否则越小。

4)计算离群因子 lof_k^i:样本点 i 的离群因子是其邻域内其他样本点与样本点 i 的局部可达密度比值的平均值。

$$lof_k^i=\frac{\sum\limits_{j\in N_k^i}lrd_k^j}{|N_k^i|}/lrd_i \tag{4}$$

lof_k^i 表示一个样本的局部相对密度,lof_k^i 越小于或者等于 1,则说明该样本点的局部可达密度大于或者等于邻域内其他样本点的局部可达密度,表明该样本处于样本中趋于中心的位置,该样本是正常样本;lof_k^i 越大于 1,则表示该样本点的局部可达密度可能比邻域内其他样本点的局部可达密度小,该样本点越可能处于所有样本的边缘,即越可能是异常样本。为了表示方便,本文将离群因子计算形式化地描述为 $lof=LOF(D)$,D 表示样本之间距离的集合。

4 问题描述

本章分别描述了 PRFL 的系统模型、威胁模型和设计目标。

4.1 方案概述

本文方案主要包含 3 种实体:可信第三方 TP、聚合服务器 S_1,S_2 和用户。

可信第三方 TP:TP 可以是企业本身或者是监管机构,负责管理整个联邦学习任务,在任务初始化阶段会将公共参数下发给聚合服务器和用户。

聚合服务器:聚合服务器负责接收用户发送的本地模型,并通过交互协同排除投毒模型,确保全局模型有效聚合。聚合服务器 S_1 和 S_2 是两个对等的实体,可以由不同的云服务器提供商提供,协同完成联邦学习聚合任务。

用户:一个用户代表一个客户端,它可以是移动手机、物联网设备等,拥有一定数量的本地数据。

4.2 威胁模型

本文考虑聚合服务器是半诚实且好奇的,即它们会遵循既定的协议,但是会对用户的隐私感到好奇,企图用户提交的

信息窃取用户隐私数据。此外，本文还假设它们是不共谋且不与用户共谋的。例如，聚合服务器由不同的云服务提供商提供，共谋会影响云服务提供商的利益和信誉。这种假设已经被广泛应用在联邦学习中[21-22,37-38]。

对于用户，PRFL 同时考虑到诚实用户、半诚实用户和拜占庭用户。其中，诚实用户和半诚实用户会遵守既定的协议进行训练，但半诚实用户会对其他用户的隐私数据感兴趣，企图从收到的模型数据中推断出诚实用户隐私数据。此外，它们也可能通过共谋的方式来提高攻击成功率。拜占庭用户则企图通过提交投毒模型来破坏全局模型的可用性或在全局模型中嵌入后门。一个半诚实用户同时也可能是拜占庭用户。本文假设在一轮训练过程中至少有两个诚实用户参与训练，即 $|\mathcal{U}|-|\mathcal{M}|\geqslant 2$，其中，$\mathcal{M}$ 表示半诚实用户集合。

4.3　设计目标

本文提出的方案旨在能够实现联邦学习模型隐私保护聚合，同时还可以抵抗拜占庭用户的投毒攻击。主要目标如下。

准确性：在没有攻击时，PRFL 需要保证全局模型的准确率在合理的范围内。

隐私性：PRFL 需要保证半诚实且好奇的用户和聚合服务器只能获取到诚实用户的正常输出，不能获取到额外的任何隐私信息。

鲁棒性：受拜占庭用户提交的投毒模型的影响，全局模型的准确率在合理范围内。

高效性：考虑到跨设备场景下联邦学习客户端资源受限的情况，PRFL 不会给用户引入不可忽略的计算开销和通信开销。

5　方案设计

5.1　总体描述

为了实现隐私保护聚合，同时可以抵抗拜占庭用户的投毒攻击，本文设计一种双服务器结构下隐私保护模型鲁棒聚合方法，如图 1 所示。

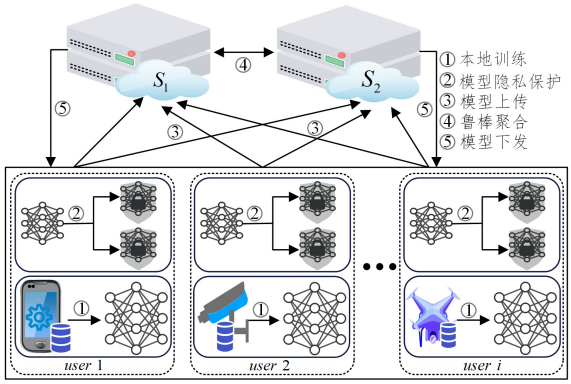


图 1　隐私保护鲁棒聚合

Fig. 1　Privacy-preserving robust aggregation

为了保护模型隐私，用户将本地模型划分为子模型并加密，然后发送给聚合服务器参与聚合。由于模型被加密，因此外界敌手和半诚实服务器在不知道密钥的情况下无法获取原始模型，从而保证了本地模型的隐私性。为了能够抵抗拜占庭用户，本文提出了一种密态模型距离计算方法，使服务器能够安全地计算出用户模型间的距离，然后基于距离值和 LOF 算法检测投毒模型，实现鲁棒聚合。综上，本方案可以实现：1）半诚实且好奇的聚合服务器无法学习到除了用户的输出外的隐私信息；2）存在拜占庭用户的情况下，能够聚合出有效的全局模型。一轮训练过程主要包含以下几个阶段。

本地训练：用户在收到初始全局模型后，使用本地数据集进行训练，得到本地模型 w_i。

模型保护：用户将 w_i 划分成两个子模型 $[w_i]^1$ 和 $[w_i]^2$，并分别进行加密，然后将加密的子模型分别发送给 S_1 和 S_2。

鲁棒聚合：S_1 和 S_2 收到子模型之后，通过密态模型距离计算方法协同计算模型距离，然后基于模型距离，利用离群因子检测投毒模型并将其丢弃。最后，S_1 和 S_2 分别聚合剩余的子模型并将结果发送给用户，用户在本地聚合出全局模型。

5.2　PRFL

5.2.1　初始化

在联邦学习训练开始前，用户和服务器收到可信第三方下发的公共参数，包括但不限于初始全局模型 w、大素数 p、生成元 g、伪随机数生成器（Pseudorandom Generator，PRG）等。之后，聚合服务器 S_1 和 S_2 分别与每个用户通过 Diffie-Hellman 密钥协商算法协商一个初始协商密钥，具体过程如下。

用户 u_i 随机选择两个不相同的整数 sk_i^1 和 sk_i^2 作为私钥，然后计算两个公钥 $g^{sk_i^1} \bmod p$ 和 $g^{sk_i^2} \bmod p$。

S_1 和 S_2 分别随机选择一个整数 sk_{S_1} 和 sk_{S_2} 作为私钥，然后计算公钥 $g^{sk_{S_1}} \bmod p$ 和 $g^{sk_{S_2}} \bmod p$。

每个用户 u_i 分别与 S_1 和 S_2 进行密钥协商，计算协商密钥 $sk_{i,S_1}=g^{sk_{S_1}sk_i} \bmod p$ 和 $sk_{i,S_2}=g^{sk_{S_2}sk_i} \bmod p$。

初始化结束，每个用户 u_i 分别拥有两个初始协商密钥 sk_{i,S_1} 和 sk_{i,S_2}。

5.2.2　模型隐私保护

用户 u_i 利用本地数据集 $\mathcal{D}_i$ 通过随机梯度下降算法训练全局模型，得到本地模型 w_i：

$$w_i=\Delta\mathcal{L}_f(w,\mathcal{D}_i) \tag{5}$$

本地模型参数包含了本地训练数据的特征信息。为了防止半诚实且好奇的聚合服务器通过模型参数窃取用户数据隐私，需要对模型参数进行保护，如算法 1 所示。

算法 1　本地模型保护

输入：$w_i, sk_{i,S_1}, sk_{i,S_2}, i\in\mathcal{U}$

输出：$[[w_i]]^1_{m\frac{n}{2}}, [[w_i]]^2_{m\frac{n}{2}}$

1. / * 第 τ 轮迭代 * /
2. for i ← 1 to $|\mathcal{U}|$ parallel do
3. $\quad [w_i]^1_{m\frac{n}{2}}[w_i]^2_{m\frac{n}{2}} \leftarrow [w_i]_{mn}$;
4. $\quad sk^{\tau}_{i,s_1}=KDF(sk_{i,S_1})$;
5. $\quad sk^{\tau}_{i,s_2}=KDF(sk_{i,S_2})$;
$\quad$ / * 生成随机掩码掩盖子模型 * /
6. $\quad [[w_i]]^1_{m\frac{n}{2}}=[w_i]^1_{m\frac{n}{2}}+PRG(sk^{\tau}_{i,s_2})$;
7. $\quad [[w_i]]^2_{m\frac{n}{2}}=[w_i]^2_{m\frac{n}{2}}+PRG(sk^{\tau}_{i,s_1})$;
8. User i send $[[w_i]]^1_{m\frac{n}{2}}$ and $[[w_i]]^2_{m\frac{n}{2}}$ to S_1 and S_2 respectively.

假设本地模型 $\boldsymbol{w}_i$ 是一个 mn 阶矩阵，用户 u_i 先将 $\boldsymbol{w}_i$ 划分

为两个子模型：

$$[w_i]^1_{m\frac{n}{2}} \| [w_i]^2_{m\frac{n}{2}} \leftarrow [w_i]_{mn}, \tag{6}$$

其中，$\|$ 表示拼接。

然后，用户 u_i 分别对子模型进行加密。为了避免给资源受限用户（例如智能手机）引入过多的计算开销，本文基于 One-Time Pad 中一次一密的思想，通过生成随机掩码掩盖的方式保护模型。相比利用其他加密方式保护模型[22]，这种方式更加高效，特别是在处理可能包含数百万个参数的模型时，掩盖模型参数不会引入过多的计算开销和通信开销。

具体来说，用户 u_i 使用初始协商密钥 sk_{i,s_1} 和 sk_{i,s_2}，通过密钥派生函数（Key Derivation Function，KDF）派生出两个临时密钥 sk^τ_{i,s_1} 和 sk^τ_{i,s_2}，然后将 sk^τ_{i,s_1} 和 sk^τ_{i,s_2} 作为种子，利用 PRG 生成随机掩码 $r_{i,s_l}=PRG(sk^\tau_{i,s_l})$，$l\in[1,2]$，并掩盖子模型。

$$[[w_i]]^1_{m\frac{n}{2}}=[w_i]^1_{m\frac{n}{2}}+PRG(sk^\tau_{i,s_2}) \tag{7}$$

$$[[w_i]]^2_{m\frac{n}{2}}=[w_i]^2_{m\frac{n}{2}}+PRG(sk^\tau_{i,s_1}) \tag{8}$$

最后，用户 u_i 分别将 $w_i{}^1_{m\frac{n}{2}}$ 和 $w_i{}^2_{m\frac{n}{2}}$ 发送给 S_1 和 S_2。S_1 和 S_2 由于只拥有自己和用户 u_i 协商的密钥，因此无法观察到原始模型参数。如果每一轮用户采用相同的种子生成相同的掩码，在模型参数范围一定的情况下服务器可能很容易猜测出原始模型参数。因此，每一轮中用户和服务器通过 KDF 派生出不同的密钥作为种子，从而生成不同的随机掩码。这样也可以避免用户每轮都需要和服务器进行密钥协商，保证用户和服务器每轮训练只需要一轮交互。

通过结合随机掩码和 KDF，PRFL 不会给用户引入过多的计算和通信开销。此外，PRFL 可以支持用户随时退出。由于掩码是由用户和服务器的协商密钥生成的，在计算模型距离时不需要用户提供，因此用户在提交模型之后退出训练，也不会影响模型聚合和对投毒模型的检测。

综上，PRFL 可以适用于参与训练的设备多为资源受限的移动设备或物联网设备的跨设备联邦学习[3]中。相比传统的联邦学习，PRFL 不会给客户端设备引入过多的计算开销以及通信开销，还可以容忍设备由于网络状态、电源等问题而频繁退出训练的问题。

5.2.3 鲁棒聚合

由于模型被加密，因此拜占庭用户可以更加隐蔽地实施投毒攻击来破坏全局模型的可用性。为了能够抵抗拜占庭用户发起的投毒攻击，需要在聚合之前对用户上传的模型进行评估，然后选择合法的本地模型进行聚合。为此，本文提出了一种密态模型距离计算方法，实现在不泄露用户子模型参数的情况下计算出用户本地模型之间的距离，具体过程如算法 2 所示。

算法 2 密态模型距离计算

输入：$[[w_i]]^1_{m\frac{n}{2}}$，$[[w_i]]^2_{m\frac{n}{2}}$，sk_{i,S_l}，sk_{i,S_2}，$i\in\mathcal{U}$

输出：$d_{i,j}$

/* 服务器 S_1 和 S_2 分别计算掩码差值并发送给对方 */

1. for l ← 1 to 2 parallel do
2. $sk^\tau_{i,s_l}=KDF(sk_{i,S_l})$；
3. $r^1_{(i,i+1)}=PRG(sk^\tau_{i,s_l})-PRG(sk^\tau_{i+1,s_l})$，$i\in\mathcal{U}$；
4. Server S_1 and S_2 send $r^1_{(i,i+1)}$，$i\in\mathcal{U}$ and $r^2_{(i,i+1)}$，$i\in\mathcal{U}$ to each other respectively；

/* 服务器 S_1 和 S_2 分别计算子模型距离 */

5. for l ← 1 to 2 parallel do
6. $r^{3-l}_{(i,j)}=\sum_i^{j-1} r^{3-l}_{i,i+1}$，$i,j\in\mathcal{U}$；
7. $d^{l'}_{i,j}=[[w_i]]^l_{m\frac{n}{2}}-[[w_j]]^l_{m\frac{n}{2}}-r^{3-l}_{i,j}$；
8. $d^l_{i,j}=\|d^{l'}_{i,j}\|_2$；
9. Server S_1 and S_2 send $d^1_{i,j}$，$i\in\mathcal{U}$ and $d^2_{i,j}$，$i\in\mathcal{U}$ to each other respectively；

/* 计算总的模型距离 */

10. Server S_1，S_2 computes total model distance：$d_{i,j}=\sqrt{(d^1_{i,j})^2+(d^2_{i,j})^2}$，$i,j\in\mathcal{U}$，$j\neq i$.

在接收到用户上传的子模型 $[w_i]^1_{m\frac{n}{2}}$ 和 $[[w_i]]^2_{m\frac{n}{2}}$ 之后，聚合服务器 S_1 和 S_2 分别利用协商密钥通过 KDF 派生出 sk^τ_{i,s_1}，sk^τ_{i,s_2}（$i\in\mathcal{U}$），然后通过 PRG 生成用户使用的随机掩码，并分别计算用户 u_i 和 u_{i+1}（$i\in\mathcal{U}$）掩码之间的差值：

$$r^1_{(i,i+1)}=PRG(sk^\tau_{i,s_1})-PRG(sk^\tau_{i+1,s_1}) \tag{9}$$

$$r^2_{(i,i+1)}=PRG(sk^\tau_{i,s_2})-PRG(sk^\tau_{i+1,s_2}) \tag{10}$$

之后，S_1 和 S_2 将得到的掩码差值发送给对方。基于 $r^1_{(i,i+1)}$，$r^2_{(i,i+1)}$（$i\in\mathcal{U}$），S_1 和 S_2 通过线性组合可以分别计算出任意两个用户 u_i 和 u_j 掩码之间的差值 $r^l_{i,j}$（$i,j\in\mathcal{U}$）：

$$r^2_{i,j}=\sum_i^{j-1}(r^2_{i,i+1}) \tag{11}$$

$$r^1_{i,j}=\sum_i^{j-1}(r^1_{i,i+1}) \tag{12}$$

接着 S_1 和 S_2 基于掩码差值可以分别计算出用户子模型参数之间的差值 $d^{1'}_{i,j}$，$d^{2'}_{i,j}$（$i,j\in\mathcal{U}$，$i\neq j$）：

$$d^{1'}_{i,j}=[[w_i]]^1_{m\frac{n}{2}}-[[w_j]]^1_{m\frac{n}{2}}-r^2_{i,j} \tag{13}$$

$$d^{2'}_{i,j}=[[w_i]]^2_{m\frac{n}{2}}-[[w_j]]^2_{m\frac{n}{2}}-r^1_{i,j} \tag{14}$$

根据子模型之间的差值，S_1 和 S_2 计算用户 u_i 和 u_j 子模型之间的欧氏距离：

$$d^1_{i,j}=\|d^{1'}_{i,j}\|_2 \tag{15}$$

$$d^2_{i,j}=\|d^{2'}_{i,j}\|_2 \tag{16}$$

其中，$\|\cdot\|$ 表示计算 l_2 范数，S_1 和 S_2 分别将 $d^1_{i,j}$，$d^2_{i,j}$（$i,j\in\mathcal{U}$）发送给对方。

S_1 和 S_2 分别收到 $d^2_{i,j}$ 和 $d^1_{i,j}$ 后，计算用户 u_i 和 u_j 模型之间的距离 $d_{i,j}$：

$$d_{i,j}=\sqrt{(d^1_{i,j})^2+(d^2_{i,j})^2} \tag{17}$$

在不向服务器泄露用户本地模型参数的情况下，S_1 和 S_2 安全计算出本地模型之间的距离 $d_{i,j}$（$i,j\in\mathcal{U}$）。

在获得模型距离的基础上，采用 3.2 节中的 LOF 算法设计了一种投毒模型检测方法。首先计算每个本地模型的离群因子，然后基于离群因子筛选并聚合良性模型，如算法 3 所示。

算法 3 模型鲁棒聚合

输入：$d_{i,j}$，$[[w_i]]^1_{m\frac{n}{2}}$，$[[w_i]]^2_{m\frac{n}{2}}$，sk^τ_{i,s_1}，sk^τ_{i,s_2} $i,j\in\mathcal{U}$

输出：w

/* 服务器计算离群因子 */

1. for l in [1,2] parallel do
2. for i in $\mathcal{U}$ do
3. $lof(w_i)=LOF([d_{i,j}])$，$j\in\mathcal{U}$；
4. $\mathcal{U}'\leftarrow[lof(w_i)]\leqslant\delta$，$i\in\mathcal{U}$；

/ * 子模型和掩码聚合 * /

5. $[[w_{s_l}]]^l_{m\frac{n}{2}}=\sum_{u_i\in\mathcal{U}'}\left(1-\frac{lof(w_i)}{\sum_{u_i\in\mathcal{U}'}lof(w_i)}\right)[[w_i]]^l_{m\frac{n}{2}}$；

6. $r_{s_l}=\sum_{u_i\in\mathcal{U}'}\left(1-\frac{lof(w_i)}{\sum_{u_i\in\mathcal{U}'}lof(w_i)}\right)PRG(sk^r_{i,s_l})$；

7. Server S_l send $[[w_{s_l}]]^l_{m\frac{n}{2}}, r_{s_l}, |\mathcal{U}'|$ to each user；

8. User i aggregates global model locally：

$$w=\frac{1}{|\mathcal{U}'|-1}([[w_{s_1}]]^1_{m\frac{n}{2}}\parallel[[w_{s_2}]]^2_{m\frac{n}{2}}-(r_{s_2}\parallel r_{s_1})).$$

S_1和S_2基于模型距离计算每个本地模型的离群因子：

$$lof(w_i)=LOF([d_{i,j}]) \tag{18}$$

由3.2节可知，lof越小，表示当前模型处于其他模型的中心；lof越大，则越有可能是投毒模型。因此，S_1和S_2选择$lof\leqslant\delta$的模型作为良性模型，大于δ的即为投毒模型。然后，S_1和S_2基于离群因子分别将良性子模型和掩码进行加权聚合：

$$[[w_{s_l}]]^l_{m\frac{n}{2}}=\sum_{u_i\in\mathcal{U}'}\left(1-\frac{lof(w_i)}{\sum_{u_i\in\mathcal{U}'}lof(w_i)}\right)[[w_i]]^l_{m\frac{n}{2}} \tag{19}$$

$$r_{s_l}=\sum_{u_i\in\mathcal{U}'}\left(1-\frac{lof(w_i)}{\sum_{u_i\in\mathcal{U}'}lof(w_i)}\right)PRG(sk^r_{i,s_l}) \tag{20}$$

其中，$l\in[1,2]$，$\mathcal{U}'$表示判定为良性模型的用户集合。最后，S_1和S_2分别将$[[w_{s_1}]]^1_{m\frac{n}{2}}, r_{s_1}, [[w_{s_2}]]^2_{m\frac{n}{2}}, r_{s_2}, |\mathcal{U}'|$发送给用户，用户在本地聚合出全局模型：

$$w=\frac{1}{|\mathcal{U}'|-1}([[w_{s_1}]]^1_{m\frac{n}{2}}\parallel[[w_{s_2}]]^2_{m\frac{n}{2}}-(r_{s_2}\parallel r_{s_1})) \tag{21}$$

联邦学习中，随着训练轮数的增加，良性模型通常往相同的方向收敛，良性模型之间的距离$d_{i,j}$会逐渐变小，良性模型w_i^{begine}的局部可达密度lrd_i与其邻域内良性模型w_j^{begine}，$j\in N_k^i$的局部可达密度lrd_j，$j\in N_k^i$相近。根据式(4)，其离群因子$lof(w_i^{\text{begine}})$会小于等于1，而投毒模型w_p^{poison}为了达到阻止模型收敛等目的，模型参数总会偏离其他的良性模型，导致其与其他模型之间的距离变大。根据3.2中可达距离的定义，这会导致w_p^{poison}与其他模型的可达距离$rd_k(p,q)$变大，最后使lrd_p局部可达密度变小。但其邻域内存在的良性模型的局部可达密度会变大，根据式(4)，投毒模型的离群因子会变大。

因此，离群因子越小，说明该模型的密度等于或者大于其周围模型密度，则该模型是良性模型；离群因子越大，则该模型的局部可达密度小于其邻域模型的局部可达密度，其越有可能是投毒模型。在IID设置下，由于用户本地数据分布相同，受到同一全局模型的影响，每一轮迭代本地模型之间的差异较小，模型收敛趋于一致，因此可以以1为阈值来筛选投毒模型。而在Non-IID的联邦学习中，用户本地数据分布的差异会导致每轮迭代模型参数之间存在一定差异，进而导致模型之间的距离增大。但每轮用户以相同的全局模型进行训练，随着训练轮数的增加，本地模型收敛方向会基本趋于一致。因此，可以通过适当地调整阈值δ，使其适用于Non-IID场景。

6　安全性分析

安全性主要保证在训练过程中，半诚实敌手无法推断出除了诚实用户正常输出值之外关于诚实用户的隐私信息。本文分别从两个方面进行了证明，在用户和服务器的交互过程中以及两个聚合服务器的交互过程中，聚合服务器和半诚实用户无法获取到关于诚实用户的隐私信息。

定理1　在用户和服务器交互过程中，S_1、S_2和半诚实用户无法获取到关于诚实用户的隐私信息。

证明：每个用户u_i使用与S_1和S_2的协商的密钥sk_{i,S_1}，sk_{i,S_2}生成子密钥，然后将子密钥作为种子通过PRG算法生成随机掩码，并分别掩盖子模型$[w_i]^2_{m\frac{n}{2}}$和$[w_i]^1_{m\frac{n}{2}}$，最后发送给S_1和S_2。首先，Computational Diffie-Hellman(CDH)问题保证了在已知G, g, g^a, g^b的情况下，计算g^{ab}是困难的。因此，在S_1和S_2不共谋的安全假设下，它们无法计算出对方和用户的协商密钥$sk_{i,S_1}, sk_{i,S_2}, i\in\mathcal{U}$。基于PRG的安全性保证，$r_{i,s_1}, r_{i,s_2}$是均匀且随机的，$S_1$和$S_2$在不知道种子的前提下，无法通过被加密的子模型计算出原始模型参数。

所有用户接收服务器返回的信息后，在本地聚合出全局模型w。首先假设没有拜占庭用户存在并且所有用户的模型都参与了聚合，由于$|\mathcal{U}|-|\mathcal{M}|\geqslant 2$，因此即使$|\mathcal{U}|-2$个用户共谋，$\mathcal{M}$中的用户通过全局模型$w$计算$\tilde{w}=w-\sum_{i\in\mathcal{U}}w_i$，每个诚实用户的本地模型也无法被准确计算。当$\mathcal{M}$中存在拜占庭用户时，其提交的投毒模型可能没有参与聚合。基于聚合服务器和用户不共谋的安全假设，$\mathcal{M}$中的用户无法确定投毒模型是否参与了聚合，$\tilde{w}\neq w-\sum_{i\in\mathcal{U}}w_i$，诚实用户的本地模型仍无法被计算出来。

综上，在用户和聚合服务器交互过程中，半诚实服务器和用户无法获取到关于诚实用户的隐私信息。

定理2　两个不共谋的聚合服务器在进行交互时，诚实用户的隐私信息不会被泄露。

证明：在安全距离计算时，S_1和S_2分别计算用户u_i和u_{i+1}的掩码差值并发送给对方。假设S_1和S_2在收到对方发送的$r^2_{(i,i+2)}, r^1_{(i,i+2)}, i\in\mathcal{U}$后，企图通过该值计算出用户的随机掩码。以$S_1$为例，$S_1$根据$r^2_{(i,i+2)}, i\in\mathcal{U}$可以构造出$n-1$个线性方程：

$$x_1-x_2=r^2_{(1,2)} \tag{22}$$

$$\cdots$$

$$x_{n-1}-x_n=r^2_{(n-1,n)} \tag{23}$$

其中，$x_1,\cdots,x_n$分别代表$PRG(sk_{1,s_2}),\cdots,PRG(sk_{n,s_2})$。线性方程组的系数矩阵和增广矩阵分别为：

$$\begin{bmatrix}1&-1&0&\cdots&0\\0&1&-1&\cdots&0\\&&&\cdots&\\0&0&0&\cdots&-1\end{bmatrix} \tag{24}$$

$$\begin{bmatrix}1&-1&0&\cdots&0&r^2_{(1,2)}\\0&1&-1&\cdots&0&r^2_{(2,3)}\\&&&\cdots&&\\0&0&0&\cdots&-1&r^2_{n-1,n}\end{bmatrix} \tag{25}$$

可以看到，其系数矩阵和增广矩阵的秩均为$n-1<n$。根据线性方程组解的判定定理，该线性方程组有解且有无数解，所以，S_1无法确定地计算出$PRG(sk_{1,s_2}),\cdots,PRG(sk_{n,s_2})$。

反之亦然，S_2 也无法确定地计算出 $PRG(sk_{1,s_1})$，…，$PRG(sk_{n,s_1})$。

综上，在 S_1 和 S_2 不共谋的情况下，S_1 和 S_2 在进行安全距离计算过程中，诚实用户的隐私信息不会被泄露。

7 实验和评估

实验将从两个角度分析 PRFL 的性能：1）评估 PRFL 在没有恶意用户且数据集为独立同分布和非独立同分布设置下的准确率；2）评估 PRFL 对不同投毒攻击（目标投毒和非目标投毒）的鲁棒性。此外，为了更好地评估 PRFL 的性能，将 PRFL 和 FedAvg 以及其他鲁棒聚合方法进行对比。

7.1 实验设定

实验采用 Python 实现，运行在 Ubuntu18.04 操作系统中，CPU 为 Intel(R) Xeon(R) Gold 5218 CPU @ 2.30 GH，内存为 256 GB。机器学习框架采用 Pytorch，使用 AES 的 CTR 模式作为伪随机生成器。

本文采用 MNIST 和 CIFAR10 数据集进行实验。其中，MNIST 数据集包含 70 000 张从 0 到 9 的手写数字图像。图像为灰度图，大小为 28×28 像素，分为 60 000 个训练样本和 10 000 测试样本。CIFAR10 数据集由 60 000 幅彩色图像组成，包含 10 个类别。数据集包含 50 000 个训练样本和 10 000 个测试样本。本文采用 CNN 模型进行训练，其中，MNISTCNN 包含 2 个卷积层、2 个池化层以及 2 个全连接层，CIFARCNN 包含 2 个卷积层、2 个池化层以及 3 个全连接层，卷积核大小均为 5。激活函数为 ReLU 和 log_softmax，损失函数为交叉熵损失函数，优化算法采用随机梯度下降算法。其他训练参数设置如表 1 所列。

表 1 参数设置

Table 1 Parameter settings

	MNIST	Cifar10
用户数量	100	100
学习率	0.001	0.01
Momentum	0.9	0.5
Local Epochs	3	5
Batch Size	64	32
Iterations	100	200

针对上述两种数据集，分别考虑了数据是 IID 和 Non-IID 的场景。在数据 IID 中，训练样本被均匀地分给每个用户。在数据 Non-IID 的情况下，每个用户的样本包含两个类别。针对这两种数据分布，分别将 δ 值设置为 1 和 1.5，邻域 k 为 70。

本文同时考虑了无目标攻击和目标攻击。无目标攻击包括高斯攻击、符号翻转攻击[29]；目标攻击是标签翻转攻击。

高斯攻击：攻击者在上传的模型参数中添加高斯噪声来阻止模型收敛。具体来说，攻击者生成均值为 0、标准偏差为 0.5 的高斯噪声并将其添加到模型参数中。

符号翻转攻击：攻击者会翻转梯度或者模型参数的符号并放大其大小。具体来说，假设攻击者的训练结果的真实值为 w_b，它会发送 $\sigma \cdot w_b$ 给聚合服务器。σ 是一个负常数，本文中 σ 设置为 -1。

标签翻转攻击：攻击者将目标训练样本的标签从一个类（Source Class）改为另一个类（Target Class），然后基于修改后的数据集进行训练。具体来说，在 MNIST 中，攻击者将样本中的标签 7 改为 1；Cifar10 中，攻击者将样本中标签“Cat”改为“Dog”。

本文根据投毒攻击的目的，分别在测试集中采用以下评估指标来评估和衡量 PRFL 的性能。

准确性（Acc）：预测正确的样本数量除以测试集样本总数。

源样本准确性（SRC-Acc）：目标样本（带有源标签）在测试集中的准确率。

攻击成功率（ASR）：将目标样本（带有源标签）错误分类为 Target Class 的比例。

为了更好地评估 PRFL 的性能，将其与经典的鲁棒聚合方案进行了比较。

Krum[17]：基于欧氏距离聚合，选择与其他 $|\mathcal{U}|-|\mathcal{M}|-1$ 个本地模型距离最近的模型作为新的全局模型。

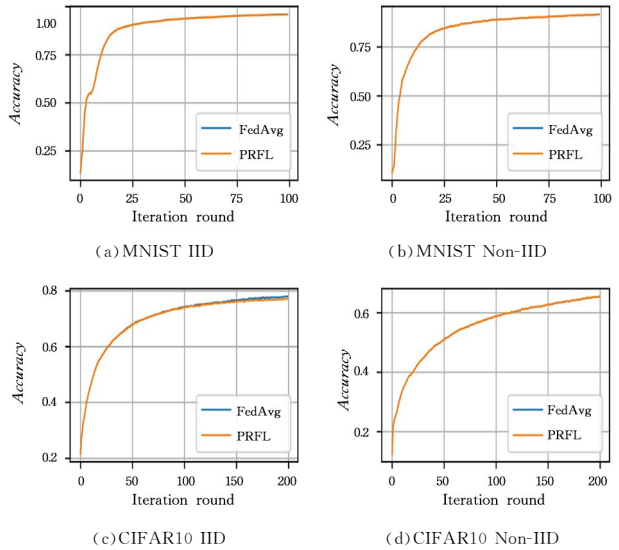
Median[18]：基于中位数聚合，先对本地模型每一维参数排序，再选取每一维的中位数作为新的全局模型参数。

Trimmed mean[18]：基于裁剪聚合，对本地模型的每一维参数排序，然后丢弃掉最大和最小的一部分值，取剩余值的平均值作为全局模型参数。

7.2 实验结果分析

7.2.1 准确性

为了验证 PRFL 的准确性，在没有拜占庭用户时，本文分别在 MNIST 和 CIFAR10 数据集的 IID 和 Non-IID 设置下进行实验，并将实验结果与基线 FedAvg[1] 进行对比。训练过程中全局模型准确率的变化情况如图 2 所示。



(a) MNIST IID (b) MNIST Non-IID

(c) CIFAR10 IID (d) CIFAR10 Non-IID

图 2 FedAvg[1] 和 PRFL 的准确率

Fig. 2 Accuracy of FedAvg[1] and PRFL

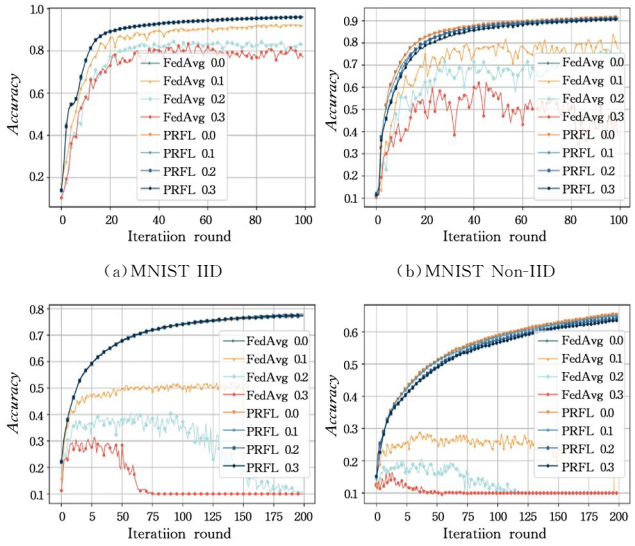
实验结果表明，无论数据是 IID 还是 Non-IID，在训练过程中，PRFL 的准确率基本与 FedAvg 保持一致。这说明 PRFL 实现了准确性目标。

7.2.2 鲁棒性

为了验证 PRFL 的鲁棒性，分别在拜占庭用户比例为

10%，20%和 30%的情况下通过 6.1 节中的指标来评估方案，同时将结果与 FedAvg 进行比较来评估 PRFL 对投毒攻击的鲁棒性。

针对无目标攻击的鲁棒性：高斯攻击和符号翻转攻击属于无目标攻击，其目标是通过阻止全局模型收敛使联邦学习无法聚合出有效全局模型。首先，通过 Acc 来评估 PRFL 的性能并和 FedAvg 比较，结果如图 3 和图 4 所示。

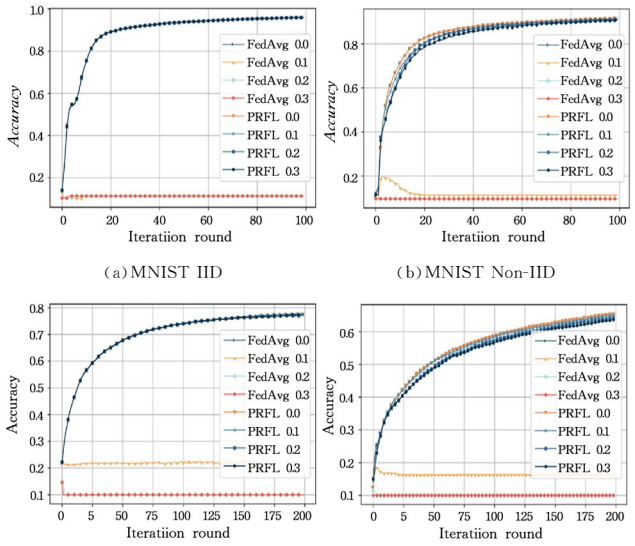


(a) MNIST IID　(b) MNIST Non-IID

(c) CIFAR10 IID　(d) CIFAR10 Non-IID

图 3　FedAvg[1] 和 PRFL 在高斯攻击下拜占庭用户数量对全局模型的影响

Fig. 3　Impact of the number of Byzantine users on global model with FedAvg[1] and PRFL under Gaussian attack



(a) MNIST IID　(b) MNIST Non-IID

(c) CIFAR10 IID　(d) CIFAR10 Non-IID

图 4　FedAvg[1] 和 PRFL 在符号翻转攻击下拜占庭用户数量对全局模型的影响

Fig. 4　Impact of the number of Byzantine users on global model with FedAvg[1] and PRFL under sign-flipping attacks

通过观察实验结果可以发现，由于拜占庭用户实施高斯攻击和符号翻转攻击，因此相比没有拜占庭用户时，FedAvg 聚合的全局模型受到了很大的影响。特别是随着拜占庭用户数量和迭代轮数的增加，FedAvg 逐渐不能收敛或者无法聚合出有效的全局模型。而 PRFL 可以有效地抵抗高斯攻击和符号翻转攻击，无论数据是 IID 还是 Non-IID，基本都保证全局模型能够收敛并得到较高准确率。

在 IID 设置下，PRFL 聚合出的全局模型的准确率几乎达到了没有拜占庭用户时的水平，如表 2 和表 3 所列。在数据 Non-IID 时，随着拜占庭用户数量增加，最终全局模型的准确率略有下降。这是因为随着拜占庭用户数量增加，它们的模型被识别为投毒模型，无法参与聚合，这使得全局模型无法很好地学习拜占庭用户所占有的训练样本的特征，最终影响了全局模型。

表 2　高斯攻击下 PRFL 和 FedAvg[1] 的准确率

Table 2　Accuracy of PRFL and FedAvg[1] under Gaussian attack

(%)

	MNIST IID	MNIST Non-IID	CIFAR10 IID	CIFAR10 Non-IID
FedAvg(0%)	0.960	0.917	0.780	0.654
FedAvg(10%)	0.926	0.840	0.523	0.289
FedAvg(20%)	0.845	0.773	0.405	0.205
FedAvg(30%)	0.833	0.62	0.315	0.164
PRFL(10%)	0.959	0.912	0.773	0.650
PRFL(20%)	0.960	0.910	0.773	0.643
PRFL(30%)	0.959	0.906	0.775	0.635

表 3　符号翻转攻击下 PRFL 和 FedAvg[1] 准确率

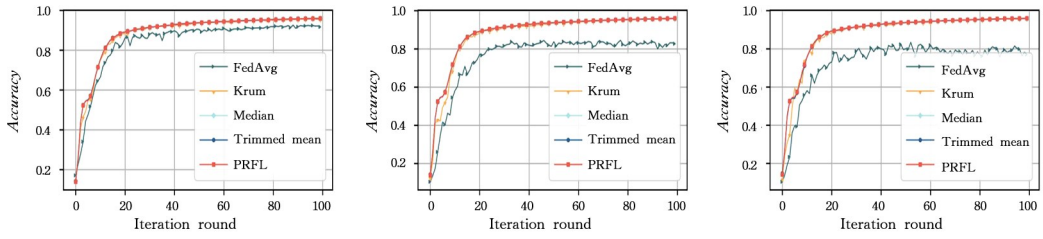
Table 3　Accuracy of PRFL and FedAvg[1] under sign-flipping attack

(%)

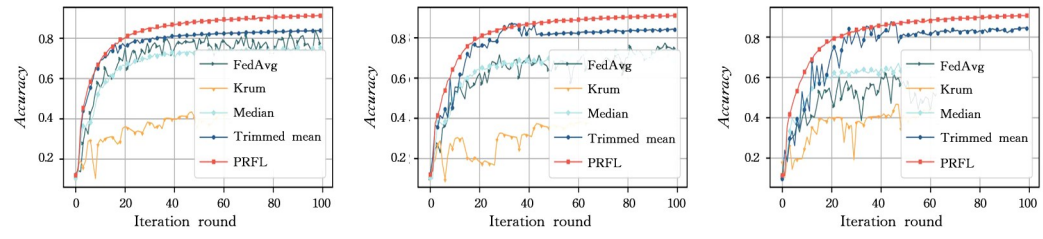
	MNIST IID	MNIST Non-IID	CIFAR10 IID	CIFAR10 Non-IID
FedAvg(0%)	0.960	0.917	0.780	0.654
FedAvg(10%)	0.130	0.199	0.226	0.185
PRFL(10%)	0.959	0.912	0.773	0.650
PRFL(20%)	0.960	0.910	0.771	0.643
PRFL(30%)	0.959	0.907	0.776	0.637

为了进一步评估 PRFL 性能，将其与现有方案(Krum，Median，Trimmed mean)进行了比较，结果如图 5－图 8 所示。通过观察发现，在 IID 设置下，无论是针对高斯攻击还是符号翻转攻击，几种聚合方案都能有效防御。其中，无论是哪种攻击和数据集，随着拜占庭用户数量的增加，Trimmed mean 和 PRFL 两种方法始终都可以达到较高的准确率。Krum 面对复杂的数据集 CIFAR10 时，虽然能够抵抗投毒攻击，但全局模型无法收敛且准确率较低。此外，在抵抗符号翻转攻击时，当拜占庭用户数量大于 20%时，Median 得到的全局模型准确率逐渐下降。

在两种数据集的 Non-IID 设置下，Krum 不能有效抵抗上述两种攻击，无法聚合出有效的全局模型。Median 和 Trimmed mean 对两种攻击具有一定的鲁棒性，但收敛速度和模型准确率受到了很大的影响，聚合出的全局模型的准确率基本都低于 PRFL。在符号翻转攻击中，拜占庭用户数量为 30%时，Median 在 CIFAR10 数据集上的准确率降到了 21.26%。而本文方案 PRFL 可以有效防御高斯攻击和符号翻转攻击，并且获得了较高的准确率。在这两种攻击下，拜占庭用户数量 30%时，PRFL 在两种数据集上的准确率分别达到了 90.6%，63.5%和 90.7%，63.7%。



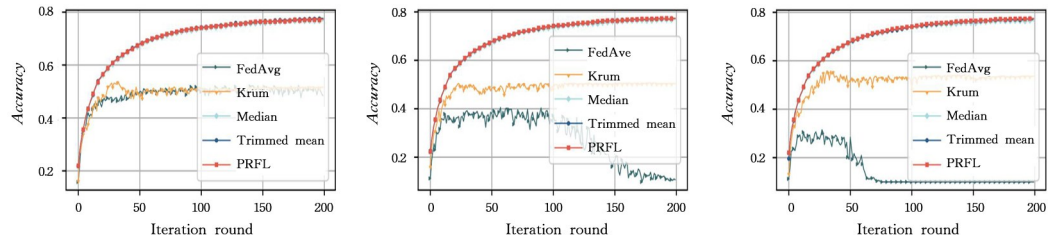
(a)MNIST IID 拜占庭用户占 10% (b)MNIST IID 拜占庭用户占 20% (c)MNIST IID 拜占庭用户占 30%



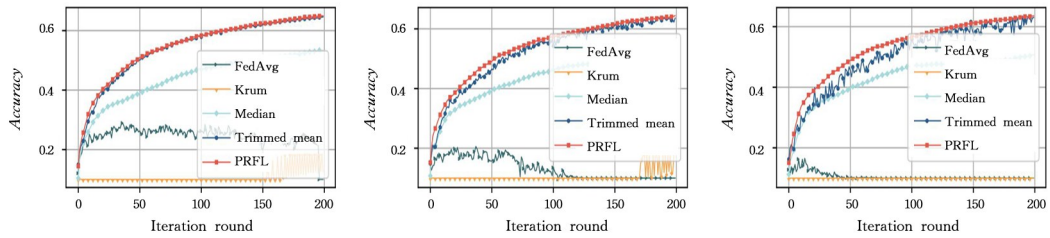
(d)MNIST Non-IID 拜占庭用户占 10% (e)MNIST Non-IID 拜占庭用户占 20% (f)MNIST Non-IID 拜占庭用户占 30%

图 5 FedAvg[1], Krum[17], Median[18], Trimmed mean[18] 和 PRFL 对高斯攻击的鲁棒性(MNIST)

Fig. 5 Robustness of FedAvg[1], Krum[17], Median[18], Trimmed mean[18] and PRFL against Gaussian attacks(MNIST)



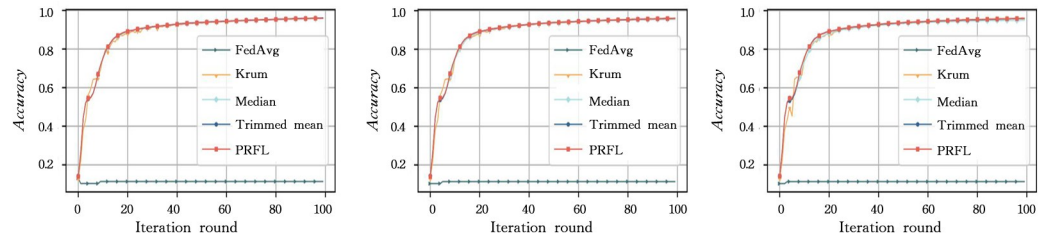
(a)CIFAR10 IID 拜占庭用户占 10% (b)CIFAR10 IID 拜占庭用户占 20% (c)CIFAR10 IID 拜占庭用户占 30%



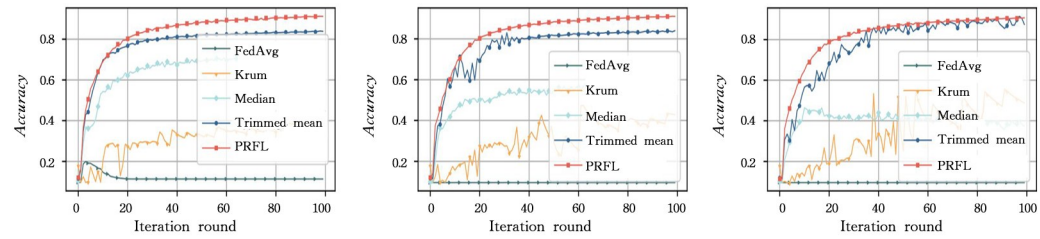
(d)CIFAR10 Non-IID 拜占庭用户占 10% (e)CIFAR10 Non-IID 拜占庭用户占 20% (f)CIFAR10 Non-IID 拜占庭用户占 30%

图 6 FedAvg[1], Krum[17], Median[18], Trimmed mean[18] 和 PRFL 对高斯攻击的鲁棒性(CIFAR10)

Fig. 6 Robustness of FedAvg[1], Krum[17], Median[18], Trimmed mean[18] and PRFL against Gaussian attacks(CIFAR10)



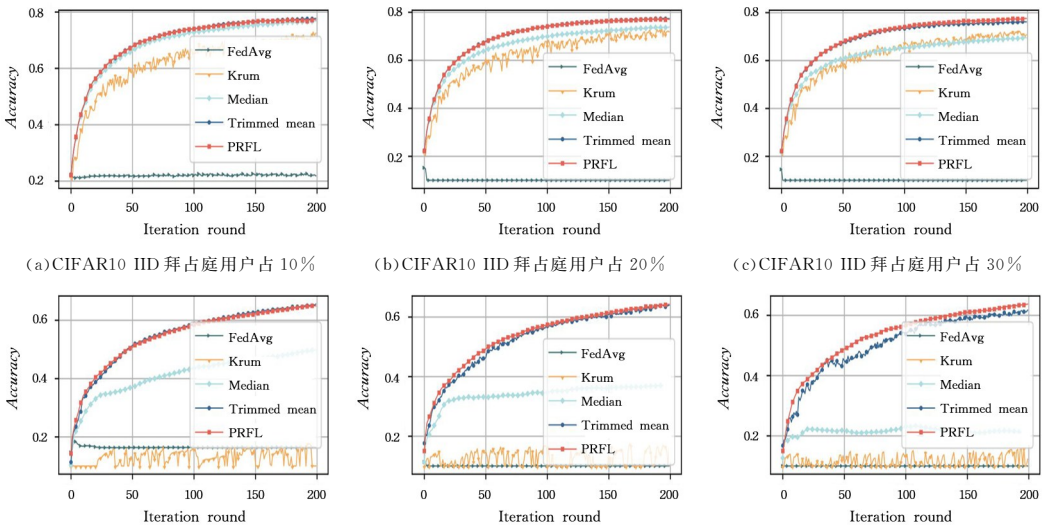
(a)MNIST IID 拜占庭用户占 10% (b)MNIST IID 拜占庭用户占 20% (c)MNIST IID 拜占庭用户占 30%



(d)MNIST Non-IID 拜占庭用户占 10% (e)MNIST Non-IID 拜占庭用户占 20% (f)MNIST Non-IID 拜占庭用户占 30%

图 7 FedAvg[1], Krum[17], Median[18], Trimmed mean[18] 和 PRFL 对符号翻转攻击的鲁棒性(MNIST)

Fig. 7 Robustness of FedAvg[1], Krum[17], Median[18], Trimmed mean[18] and PRFL against sign-flipping attacks(MNIST)



(a)CIFAR10 IID 拜占庭用户占 10%　(b)CIFAR10 IID 拜占庭用户占 20%　(c)CIFAR10 IID 拜占庭用户占 30%

(d)CIFAR10 Non-IID 拜占庭用户占 10%　(e)CIFAR10 Non-IID 拜占庭用户占 20%　(f)CIFAR10 Non-IID 拜占庭用户占 30%

图 8　FedAvg[1]，Krum[17]，Median[18]，Trimmed mean[18]和 PRFL 对符号翻转攻击的鲁棒性(CIFAR10)

Fig. 8　Robustness of FedAvg[1], Krum[17], Median[18], Trimmed mean[18] and PRFL against sign-flipping attacks(CIFAR10)

以上实验结果表明，PRFL 对高斯攻击和符号翻转攻击具有较强的鲁棒性，并且不受数据分布的影响，在数据 IID 和 Non-IID 设置下都能够较好地检测出投毒模型，实现鲁棒聚合。

针对目标攻击的鲁棒性：标签翻转攻击属于有目标攻击，其目的是增加目标样本被分类为 Target-class 的比例。我们增加了 Src-Acc 和 ASR 两个指标来评估聚合方案对标签翻转攻击的鲁棒性。在数据独立同分布的情况下，分别在 MNIST 和 CIFAR10 数据集进行实验，拜占庭用户比例为 30%。将无攻击下的 FedAvg 作为基线，同时将实验结果和其他鲁棒聚合方案进行比较。Src-Acc 变化如图 9 和图 10 所示。

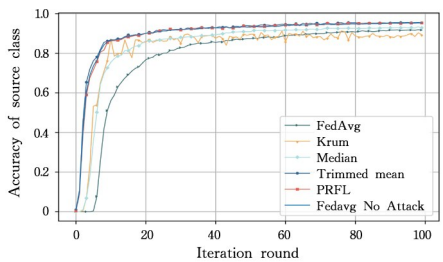


图 9　FedAvg[1]，Krum[17]，Median[18]，Trimmed mean[18]和 PRFL 对标签翻转攻击的鲁棒性(MNIST)

Fig. 9　Robustness of FedAvg[1], Krum[17], Median[18], Trimmed mean[18] and PRFL against label flipping attacks(MNIST)

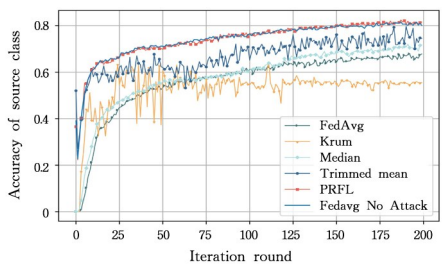


图 10　FedAvg[1]，Krum[17]，Median[18]，Trimmed mean[18]和 PRFL 对标签翻转攻击的鲁棒性(CIFAR10)

Fig. 10　Robustness of FedAvg[1], Krum[17], Median[18], Trimmed mean[18] and PRFL against label flipping attacks(CIFAR10)

可以看到，无论是 MNIST 还是 CIFAR10 数据集，Trimmed mean，Median 和 PRFL 都可以降低标签翻转攻击的影响。其中，在 MNIST 数据集中，Trimmed mean 和 PRFL 基本可以消除标签翻转攻击的影响；而在 CIFAR10 数据集中，只有 PRFL 可以很好地抵抗标签翻转攻击。在两种数据集下，Krum 都不能有效抵抗标签翻转攻击。

表 4 列出了标签翻转攻击下各聚合方案的 ACC，Src-Acc 以及 ASR。可以看到，标签翻转攻击没有显著降低 Acc，但是会增加目标样本被识别为 Target-class 的比例，降低 Src-Acc。Krum，Median，Trimmed mean 以及 PRFL 相比 FedAvg，都可以有效降低 ASR。但 Krum 获得了较低的 Acc 和 Src-Acc，这显然是不能接受的。在 MNIST 数据集中，Trimmed mean 和 PRFL 可以将 ASR 降低到与基线一样，同时保证了 Acc 和 Src-Acc 几乎不受影响。在 CIFAR10 数据集下，Trimmed mean，Median 和 PRFL 都能够有效缓解符号翻转攻击带来的影响，其中 PRFL 的 Acc，Src-Acc 以及 ASR 都是最优的。

表 4　标签翻转攻击下不同鲁棒聚合方法的性能

Table 4　Performance of various robust aggregation methods under label flipping attack

	MNIST			CIFAR10		
	Acc/%	*Src-Acc*/%	*ASR*	*Acc*/%	*Src-Acc*/%	*ASR*
FedAvg[1](No Attack)	**96.03**	95.33	**0.68**	**77.85**	80.1	**0.4**
FedAvg[1]	95.61	91.73	2.33	76.92	67.8	4.1
Krum[17]	94.31	89.2	0.97	50.55	55.5	2.2
Median[18]	95.69	92.90	1.56	77.17	71.8	1.5
Trimmed mean[18]	95.94	**95.43**	**0.68**	76.96	74.5	1.0
PRFL	**95.97**	95.33	**0.68**	**77.57**	**81.00**	**0.7**

综上，相比其他鲁棒聚合方法，PRFL 可以有效防御标签翻转攻击，同时还能够保持良好的模型准确率。

上述实验证明，PRFL 实现了鲁棒性的目标，并且在大多数情况下其表现优于现有的 Krum，Median，Trimmed mean 方法。

结束语　针对联邦学习容易同时受到投毒攻击和隐私

攻击的问题，本文提出了一种隐私保护联邦学习鲁棒聚合方法 PRFL。该方法能够有效抵抗拜占庭用户实施的投毒攻击，同时能够保护用户模型隐私。在两种图像数据集上的广泛实验证明，PRFL 对多种类型投毒攻击都具有较好的鲁棒性，同时还保证了模型准确率。无论是 IID 设置还是 Non-IID 设置，PRFL 都优于现有的 Krum，Median 和 Trimmed mean 方法。此外，PRFL 不会给用户带来计算开销和通信开销问题。

在未来的工作中，我们将重点研究针对更加复杂的投毒攻击的有效防御措施以及如何在更强大的敌手模型下保护模型隐私。

参 考 文 献

[1] MCMAHAN B, MOORE E, RAMAGE D, et al. Communication-Efficient Learning of Deep Networks from Decentralized Data[C]// Proceedings of the 2017 International Conference on Artificial Intelligence and Statistics. Brookline: Microtome Publishing, 2017: 1273-1282.

[2] VOIGT P, BUSSCHE A V D. The Eu General Data Protection Regulation(GDPR)[M]. Berlin: Springer, 2017: 1-383.

[3] KAIROUZ P, MCMAHAN H B, AVENT B, et al. Advances and Open Problems in Federated Learning[J]. Foundations and Trends in Machine Learning, 2021, 14(1/2): 1-210.

[4] MOTHUKURI V, PARIZI R M, POURIYEH S, et al. A Survey on Security and Privacy of Federated Learning[J]. Future Generation Computer Systems-the International Journal of Escience, 2021, 115: 619-640.

[5] SHOKRI R, STRONATI M, SONG C Z, et al. Membership Inference Attacks against Machine Learning Models[C]// Proceedings of the 2017 IEEE Symposium on Security and Privacy. New York: IEEE Press, 2017: 3-18.

[6] ZHU L G, LIU Z J, HAN S. Deep Leakage from Gradients [C]// Proceedings of the 2019 International Conference on Neural Information Processing Systems. Los Angeles: NIPS, 2019: 1323-1334.

[7] SALEM A, ZHANG Y, HUMBERT M, et al. ML-Leaks: Model and Data Independent Membership Inference Attacks and Defenses on Machine Learning Models[C]// Proceedings of the 2019 Network and Distributed System Security Symposium. Reston: Internet Society, 2019: 1-15.

[8] GEIPING J, BAUERMEISTER H, DRÖGE H, et al. Inverting Gradients-How Easy Is It to Break Privacy in Federated Learning? [C]// Proceedings of the 2020 International Conference on NeuralInformation Processing Systems. Los Angeles: NIPS, 2020: 16937-16947.

[9] 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.

[10] WEI K, LI J, DING M, et al. Federated Learning with Differential Privacy: Algorithms and Performance Analysis[J]. IEEE Transactions on Information Forensics and Security, 2020, 15: 3454-3469.

[11] ZHOU H, YANG G, HUANG Y, et al. Privacy-Preserving and Verifiable Federated Learning Framework for Edge Computing [J]. IEEE Transactions on Information Forensics and Security, 2023, 18: 565-580.

[12] STEVENS T, SKALKA C, VINCENT C, et al. Efficient Differentially Private Secure Aggregation for Federated Learning Via Hardness of Learning with Errors[C]// Proceedings of the 2022 USENIX Security Symposium. Boston: USENIX Association, 2022: 1379-1395.

[13] MA J, NAAS S A, SIGG S, et al. Privacy-Preserving Federated Learning Based on Multi-Key Homomorphic Encryption[J]. International Journal of Intelligent Systems, 2022, 37(9): 5880-5901.

[14] PHONG L T, AONO Y, HAYASHI T, et al. Privacy-Preserving Deep Learning Via Additively Homomorphic Encryption[J]. IEEE Transactions on Information Forensics and Security, 2018, 13(5): 1333-1345.

[15] ZHU H, WANG R, JIN Y, et al. Distributed Additive Encryption and Quantization for Privacy Preserving Federated Deep Learning[J]. Neurocomputing, 2021, 463: 309-327.

[16] FUNG C, YOON C J M, BESCHASTNIKH I. The Limitations of Federated Learning in Sybil Settings[C]// Proceedings of the 2020 International Symposium on Research in Attacks, Intrusions and Defenses. USENIX Association, 2020: 301-316.

[17] BLANCHARD P, MHAMDI E M E, GUERRAOUI R, et al. Machine Learning with Adversaries: Byzantine Tolerant Gradient Descent[C]// Proceedings of the 2017 International Conference on Neural Information Processing Systems. Los Angeles: NIPS, 2017: 118-128.

[18] YIN D, CHEN Y, KANNAN R, et al. Byzantine-Robust Distributed Learning: Towards Optimal Statistical Rates[C]// Proceedings of the 2018 International Conference on Machine Learning. San Diego: JMLR, 2018: 5650-5659.

[19] SUN Z, KAIROUZ P, SURESH A T, et al. Can You Really Backdoor Federated Learning? [J]. arXiv: abs/1911. 07963, 2019.

[20] SO J, GÜLER B, AVESTIMEHR A S. Byzantine-Resilient Secure Federated Learning[J]. IEEE Journal on Selected Areas in Communications, 2021, 39(7): 2168-2181.

[21] MA Z, MA J, MIAO Y, et al. ShieldFL: Mitigating Model Poisoning Attacks in Privacy-Preserving Federated Learning[J]. IEEE Transactions on Information Forensics and Security, 2022, 17: 1639-1654.

[22] LIU X, LI H, XU G, et al. Privacy-Enhanced Federated Learning against Poisoning Adversaries[J]. IEEE Transactions on Information Forensics and Security, 2021, 16: 4574-4588.

[23] NASERI M, HAYES J, DE CRISTOFARO E. Local and Central Differential Privacy for Robustness and Privacy in Federated Learning[C]// Proceedings of the 2022 Network and Distributed System Security Symposium. Reston: Internet Society, 2022: 1-19.

[24] JEBREEL N M, DOMINGO-FERRER J, BLANCO-JUSTICIA A, et al. Enhanced Security and Privacy Via Fragmented Federated Learning[J]. IEEE Transactions on Neural Networks and Learning Systems, 2024, 35(5): 6703-6717.

[25] BREUNIG M M,KRIEGEL H P,NG R T,et al. LOF:Identifying Density-Based Local Outliers[C]// Proceedings of the 2000 ACM SIGMOD International Conference on Management of data. New York:Association Computing Machinery,2000:93-104.

[26] LIU Z,GUO J,LAM K Y,et al. Efficient Dropout-Resilient Aggregation for Privacy-Preserving Machine Learning[J]. IEEE Transactions on Information Forensics and Security,2023,18:1839-1854.

[27] JAHANI-NEZHAD T,MADDAH-ALI M A,LI S,et al. SwiftAgg:Communication-Efficient and Dropout-Resistant Secure Aggregation for Federated Learning with Worst-Case Security Guarantees[C]// Proceedings of the 2022 IEEE International Symposium on Information Theory(ISIT). Espoo:IEEE,2022:103-108.

[28] FANG M,CAO X,JIA J,et al. Local Model Poisoning Attacks to Byzantine-Robust Federated Learning[C]// Proceedings of the 2020 USENIX Security Symposium. Berkeley:USENIX Asso-ciation,2020:1623-1640.

[29] LI L,XU W,CHEN T,et al. RSA:Byzantine-Robust Stochastic Aggregation Methods for Distributed Learning from Heterogeneous Datasets[C]// Proceedings of the 2019 AAAI Conference on Artificial Intelligence Palo Alto:AAAI Press,2019:1544-1551.

[30] LI T,HU S,BEIRAMI A,et al. Ditto:Fair and Robust Federated Learning through Personalization[C]// Proceedings of the 2021 International Conference on Machine Learning. San Diego:JMLR,2021:6357-6368.

[31] BAGDASARYAN E,VEIT A,HUA Y,et al. How to Backdoor Federated Learning[C]// Proceedings of the 2020 International Conference on Artificial Intelligence and Statistics. Boston:Addison Wesley Publishing Company,2020:2938-2948.

[32] OZDAYI M S,KANTARCIOGLU M,GEL Y R. Defending against Backdoors in Federated Learning with Robust Learning Rate[C]// Proceedings of the 2021 AAAI Conference on Artificial Intelligence. Palo Alto:AAAI Press,2021:9268-9276.

[33] MA X,SUN X,WU Y,et al. Differentially Private Byzantine-Robust Federated Learning[J]. IEEE Transactions on Parallel and Distributed Systems,2022,33(12):3690-3701.

[34] CHEN X,YU H,JIA X,et al. APFed:Anti-Poisoning Attacks in Privacy-Preserving Heterogeneous Federated Learning[J]. IEEE Transactions on Information Forensics and Security,2023,18:5749-5761.

[35] XIE C,HUANG K,CHEN P Y,et al. DBA:Distributed Backdoor Attacks against Federated Learning[C]// Proceedings of the 2020 International Conference on Learning Representations. 2020:1-15.

[36] JAGIELSKI M,OPREA A,BIGGIO B,et al. Manipulating Machine Learning:Poisoning Attacks and Countermeasures for Regression Learning[C]// Proceedings of the 2018 IEEE Symposium on Security and Privacy. New York:IEEE Press,2018:19-35.

[37] MOHASSEL P,ZHANG Y. SecureML:A System for Scalable Privacy-Preserving Machine Learning[C]// Proceedings of the 2017 IEEE Symposium on Security and Privacy. New York:IEEE Press,2017:19-38.

[38] XU G W,LI H W,ZHANG Y,et al. Privacy-Preserving Federated Deep Learning with Irregular Users[J]. IEEE Transactions on Dependable and Secure Computing,2022,19(2):1364-1381.

GAO Qi, born in 1998, postgraduate. His main research interests include federated learning and privacy protection.

SUN Yi, born in 1979, Ph.D, associate professor, Ph.D supervisor. Her main research interests include network and information security, and data security exchange.

(责任编辑:柯颖)