计 算 机 研 究 与 发 展
Journal of Computer Research and Development
https://crad.ict.ac.cn
63(10): 2557−2575, 2026

面向数据异质性的联邦学习投毒防御框架

周 辉 陈 昊 秦 拯 刘 羽
(湖南大学网络空间安全学院 长沙 410082)
(huizhoucsee@hnu.edu.cn)

Poisoning Defense Framework for Federated Learning Under Data Heterogeneity

Zhou Hui, Chen Hao, Qin Zheng, and Liu Yu
(*College of Cyber Science and Technology*, *Hunan University*, *Changsha* 410082)

Abstract Federated learning (FL) enables collaborative model training without sharing raw data, making it well suited for collaborative intelligence among massive heterogeneous terminals in dynamic heterogeneous networks. However, its open participation mechanism makes FL vulnerable to poisoning attacks, where malicious clients can degrade the global model by performing label-flipping, backdoor injection, or model update manipulation. Existing poisoning defense methods primarily rely on robust aggregation strategies to mitigate malicious updates. Nevertheless, under data heterogeneity, these methods often suffer from limited adaptability, inaccurate anomaly detection caused by single-metric evaluation, and the loss of useful information due to hard filtering strategies. To address these issues, a unified poisoning defense framework for federated learning under data heterogeneity is proposed. The framework adopts ConTrust, a multidimensional consistency scoring-based defense method, as the foundation, which identifies malicious updates by jointly evaluating update direction consistency, update magnitude deviation, and principal subspace residuals. For non-independent and identically distributed (Non-IID) scenarios, HistTrust, a history behavior modeling-based defense method, is developed by incorporating local neighborhood references and cross-round reputation information, thereby mitigating the misclassification caused by data heterogeneity and improving malicious client identification. Experiments are conducted on MNIST, Fashion-MNIST, and CIFAR-10 datasets under multiple poisoning attack settings. Experimental results demonstrate that the proposed defense framework consistently reduces the attack success rate while maintaining high model accuracy under different data distributions, outperforming several representative defense methods in overall performance. The study indicates that designing differentiated defense mechanisms for different data distributions can effectively enhance the robustness of FL against poisoning attacks.

Key words federated learning; data heterogeneity; poisoning defense framework; non-independent and identically distributed; dynamic heterogeneous network environments

摘 要 联邦学习能够在不共享原始数据的条件下实现多主体协同建模,为动态异构网络中海量异质终

收稿日期: 2026-06-04; **修回日期**: 2026-07-10
基金项目: 国家自然科学基金项目(62302162, 62472166, 62441233); 国家重点研发计划(2022YFB3103500); 湖南省芙蓉计划项目(2026RC3083); 湖南省自然科学基金项目(2025JJ60414); 湖南省科技项目(2015TP1004); 湖南省研究生科研创新项目(CX20250632)
This work was supported by the National Natural Science Foundation of China (62302162, 62472166, 62441233), the National Key Research and Development Program of China (2022YFB3103500), the Science and Technology Innovation Program of Hunan Province (2026RC3083), the Natural Science Foudation of Hunan Province (2025JJ60414), the Technology Projects of Hunan Province (2015TP1004), and the Hunan Provincial Innovation Foundation For Postgraduate (CX20250632).
通信作者: 刘羽(liuyuly@hnu.edu.cn)

端的协同智能提供了有效支撑,但开放参与机制使其易遭受投毒攻击。恶意客户端可通过标签翻转、后门注入及模型更新操纵等方式破坏全局模型性能。现有联邦学习投毒防御方法多采用鲁棒聚合策略,通过识别并抑制异常更新降低恶意客户端对全局模型的影响。在数据异质性场景下,现有方法存在适应性不足、单一异常指标易误判以及硬剔除策略损失有效信息等问题。针对上述问题,提出一种面向数据异质性的联邦学习投毒防御框架。框架以多维一致性评分的防御方法 ConTrust 为基础,通过更新方向一致性、更新幅值偏离和主子空间残差实现恶意更新识别;针对非独立同分布场景,设计融合局部近邻参考与跨轮信誉信息的历史行为建模防御方法 HistTrust,缓解数据异质性导致的异常误判,提高恶意客户端识别能力。实验覆盖 MNIST、FMNIST 和 CIFAR-10 数据集及多种投毒攻击场景,结果表明,所提防御框架在不同数据分布条件下均能有效降低攻击成功率,同时保持较高模型准确率,综合防御性能优于多种典型防御方法。研究表明,针对不同数据分布特点设计差异化防御方法,能够有效提升联邦学习抵御投毒攻击的能力。

关键词 联邦学习;数据异质性;投毒防御框架;非独立同分布;动态异构网络环境

中图法分类号 TP181; TP309

DOI: 10.7544/issn1000-1239.202660415 **CSTR:** 32373.14.issn1000-1239.202660415

随着物联网、移动互联网、边缘计算等技术的快速发展,大量具备感知、计算与通信能力的终端设备持续接入网络,形成了典型的动态异构网络环境[1-2]。在智慧医疗、工业互联网、智能交通及移动智能等应用场景中,海量异质终端持续产生来源广泛、分布差异显著的数据资源,为人工智能模型训练提供了丰富的数据基础。然而,由于隐私保护需求和法律法规约束,各终端原始数据难以直接集中共享,传统集中式模型训练模式面临挑战。联邦学习(federated learning, FL)[3-4]通过“数据不出本地、数据不动模型动”的方式实现多终端协同建模,能够在保护数据隐私的同时充分利用各终端的分布式数据资源,已成为动态异构网络环境下智能协同计算的重要技术途径。

然而,联邦学习开放参与机制以及服务端无法直接获取客户端原始数据和训练过程,使其面临更加严峻的安全威胁[5-7]。其中,投毒攻击(poisoning attack)[8-10]是影响联邦学习可靠性的主要攻击方式之一。攻击者可以通过篡改本地训练数据或直接操纵模型更新等方式上传恶意更新,误导全局模型优化方向,导致模型精度下降,甚至在特定触发条件下产生攻击者预设的错误输出。由于服务器缺乏对客户端本地训练行为的直接监督能力,恶意更新往往难以及时识别,使投毒攻击具有较强的隐蔽性和持续性。因此,如何准确识别恶意更新并实现鲁棒聚合,已成为联邦学习安全研究的重要内容。

针对联邦学习投毒攻击,研究者提出了基于距离度量、鲁棒统计、相似性聚类以及历史行为分析等多种防御方法[11-19]。这些方法通常通过构建鲁棒聚合策略识别并抑制异常更新,在数据独立同分布(independent and identically distributed, IID)[20]条件下取得了较好的防御效果。然而,在动态异构网络中,不同终端由于用户行为、设备环境和业务场景差异,其本地数据通常呈现显著的数据异质性,即非独立同分布(non-independent and identically distributed, Non-IID)特征。数据异质性导致良性客户端更新之间天然存在较大的方向和幅值差异,破坏了传统防御方法所依赖的全局一致性假设,使正常更新与恶意更新之间的边界逐渐模糊,从而显著降低现有投毒防御方法的识别准确率。因此,如何提升数据异质性条件下恶意更新的识别能力,已成为联邦学习投毒防御研究面临的关键挑战。

进一步分析现有研究可以发现,其局限性主要体现在 3 个方面:1)多数方法针对单一数据分布设计,默认良性客户端更新满足一致性假设,缺乏针对不同数据分布特点构建差异化防御的能力,当数据异质性增强时容易误判正常客户端;2)现有方法通常依据单一统计特征或异常指标识别恶意更新,难以全面刻画复杂投毒攻击行为,对隐蔽攻击和协同攻击的鲁棒性不足;3)多数方法采用硬剔除策略直接舍弃可疑客户端更新,同时缺乏对客户端跨轮行为的持续建模,在恶意比例较高或数据异质性较强时容易导致有效训练信息损失,影响全局模型收敛性能。

需要指出的是,本文面向数据异质性展开防御研究,并非意味着 IID 场景不再具有研究意义。 IID 场景是数据异质性的基础情形,也是分析客户端更

新统计特征的重要基础。随着数据异质性的增强，客户端更新分布逐渐偏离统一一致性假设，传统防御策略的有效性随之下降。因此，本文旨在构建面向不同数据分布的统一投毒防御框架，实现联邦学习投毒防御从 IID 向 Non-IID 场景的自然扩展。本文的主要贡献包括 3 个方面：

1）提出面向 IID 场景的多维一致性评分防御方法 ConTrust，从方向一致性、更新幅值与子空间残差等多个维度综合评估客户端更新可信度，并通过平滑加权机制降低恶意更新对全局模型的影响；

2）提出面向 Non-IID 场景的历史行为建模防御方法 HistTrust，结合局部近邻关系与跨轮信誉信息构建客户端动态可信度，有效缓解数据异质性导致的误判问题，提高恶意客户端识别能力；

3）在多个公开数据集与典型投毒攻击场景下开展实验验证，所提框架在保持较高模型准确率的同时显著降低了攻击成功率，并在不同数据异质程度下均表现出良好的鲁棒性和泛化能力，优于现有防御方法。

1 相关工作

1.1 投毒攻击

投毒攻击是在模型训练过程中注入恶意信息，诱导模型学习错误的决策边界、错误关联关系或隐藏后门。早期研究表明，在传统机器学习与深度学习场景中，即使仅注入少量精心构造的恶意样本，也可能显著影响模型分类边界并降低模型性能[21-22]。在联邦学习中，攻击者仅需控制少量客户端，即可通过污染本地数据或操纵模型更新来影响全局模型。Tolpegin 等人[23]系统比较了标签翻转、随机标签扰动以及输入数据污染等攻击方式对联邦学习的影响，验证了数据投毒在联邦学习中的可行性与破坏能力。

根据攻击对象不同，联邦学习投毒攻击[8-10]通常可分为数据投毒和模型投毒 2 类。其中，数据投毒主要作用于客户端本地训练数据，例如标签翻转攻击将源类别标签篡改为目标类别标签，诱导本地模型学习错误映射关系；模型投毒则直接操纵客户端上传模型参数或梯度更新，包括更新放大、反向更新、方向伪造以及针对特定鲁棒聚合规则构造恶意更新等方式，通常具有更强的攻击力。Baruch 等人[24]提出“A little is enough”攻击，指出在高维梯度空间中，即使较小幅度的协同偏移也可能绕过部分鲁棒聚合规则，导致模型性能显著下降。Fang 等人[25]进一步研究了针对拜占庭鲁棒聚合算法的模型投毒攻击，证明攻击者能够依据聚合特性优化恶意更新，从而削弱距离度量类防御方法的鲁棒性。Xie 等人[26]通过内积操纵攻击突破了拜占庭容错随机梯度下降的相关假设，Shejwalkar 等人[27]从攻防博弈角度分析了模型投毒策略与防御规则之间的适配关系。

后门攻击[28-29]作为模型投毒的重要形式，其目标是在保持模型正常分类能力的同时，使携带特定触发器的样本被误分类到攻击者指定类别，因此近年来受到广泛关注。Bagdasaryan 等人[30]提出模型替换攻击，通过缩放恶意客户端模型更新，使聚合后的全局模型逼近攻击者预设的后门模型。Xie 等人[31]提出分布式后门攻击（distributed backdoor attack, DBA），将完整触发器拆分为多个局部子触发器，并由不同恶意客户端协同注入，以降低单个客户端更新的异常程度。Wang 等人[32]提出尾部分布后门攻击，利用尾部样本或稀有模式构造语义后门，无需依赖明显的人工触发图案。Neurotoxin[33]则优先扰动训练过程中变化较小的参数，使植入后门在后续正常训练中更难被覆盖，从而提升攻击持久性。

近年来，联邦学习后门攻击更强调对持久性、协同性和隐蔽性能力的提升。Chameleon[34]利用对比学习优化投毒样本与良性样本之间的表征关系，延长后门生命周期；CerP[35]从多客户端协同角度构造更加隐蔽的后门，在多种防御机制下仍能够保持较高攻击成功率；Backdoor-Critical Layers[36]则发现攻击者仅需操纵少量关键层参数即可维持后门效果。PFedBA[37]进一步指出个性化联邦学习模型同样面临后门威胁。上述研究表明，后门攻击范式已经从强扰动、高可见度转向低扰动、结构化、可持续和防御感知。由于传统联邦后门在后续良性训练轮次中容易被稀释，攻击成功率会随训练轮次增加而下降。

为增强后门持久性，Liu 等人[38]提出全组合后门攻击（full combination backdoor attack, FCBA），将全局触发器划分为多个局部触发单元，并为不同恶意客户端分配更丰富的局部触发模式，从而在聚合后形成更完整的全局后门表征。与 DBA[31]仅将触发器进行简单拆分并分散注入不同，FCBA 更强调组合触发信息的覆盖充分性和后门模式的全局完整性，其目标不仅在于提高瞬时攻击成功率，更在于增强后门在后续良性更新稀释下的攻击持久性。

1.2 投毒防御

针对联邦学习投毒攻击，现有防御方法根据恶

意更新识别依据的不同，大致可以分为基于距离度量、鲁棒统计、相似性聚类以及历史行为分析。

基于距离度量的防御方法通常假设良性客户端更新在参数空间中彼此接近，而恶意更新与多数良性更新之间存在较大偏离。Blanchard 等人[11]提出 Krum，通过计算每个客户端更新与其若干近邻之间的距离得分，并选择得分最低的更新作为聚合结果。Multi-Krum[11]则进一步选择多个候选更新后进行平均聚合。这类方法具有较好的可解释性，但其有效性依赖于良性更新在全局空间中的集中分布。当客户端数据呈现 Non-IID 时，良性客户端之间的更新差异可能显著增大，基于全局距离中心的判别机制容易将分布偏斜的正常客户端误识别为异常客户端。

基于鲁棒统计的防御方法通常不显式区分客户端身份，而是采用对异常值不敏感的统计聚合策略以减弱恶意更新的影响。Yin 等人[12]研究了坐标中值和截断均值在拜占庭环境下的统计性质，并证明其在一定条件下能够保持有界误差。El Mhamdi 等人[13]提出 Bulyan，首先利用 Multi-Krum 选择候选更新集合，再执行坐标级截断聚合，以减少高维空间中恶意更新的可利用余量。Pillutla 等人[14]提出鲁棒联邦聚合(robust federated aggregation, RFA)，使用几何中位数替代传统平均聚合，提高聚合结果对离群更新的鲁棒性。然而，坐标级或全局几何中位数方法往往忽略更新向量整体方向和主结构信息，对于低幅值方向偏移或结构性扰动的识别能力仍然有限。

基于相似性聚类的防御方法主要利用客户端更新之间的方向一致性或协同行为来识别恶意更新。Fung 等人[15]提出 FoolsGold，通过累计历史更新并计算客户端之间的余弦相似度，降低女巫式协同攻击客户端的聚合权重。Ozdayi 等人[16]提出鲁棒学习率方法，根据各维度更新符号的一致性动态调整服务器学习率，从而削弱后门更新对全局模型方向的影响。FLTrust[17]通过服务器端少量可信数据生成参考更新，并依据客户端更新与参考方向之间的一致性分配信任权重。FLAME[18]结合模型聚类、权重裁剪和自适应噪声注入，对攻击实现过滤与抑制。这类方法扩展了恶意更新检测的特征来源，但部分方法依赖可信数据或特定攻击假设，在真实场景中可能受到数据可得性和分布偏移的限制。

基于历史行为分析的防御方法通常维护客户端在多轮训练中的更新记录，并依据其历史相似性、预测一致性或信誉变化评估客户端可信度。例如，FoolsGold[15]利用历史梯度相似性抑制多个高度相似的恶意客户端；FLDetector[19]通过比较客户端当前更新与历史预测更新之间的偏差检测异常行为；FedRecover[10]则利用历史全局模型和历史客户端更新估计受污染训练过程中的正常更新，以降低攻击发生后重新训练模型的代价。此外，部分信誉驱动方法还会根据客户端在历史训练轮次中的聚合贡献或异常检测结果，动态调整其后续权重。

总体来看，现有联邦学习投毒防御方法主要围绕客户端更新异常性的刻画展开，通过距离度量、统计分析、方向一致性及历史行为等不同信息评估客户端更新可信性，并据此完成鲁棒聚合。随着研究不断深入，恶意更新识别依据逐渐由单一几何距离扩展到多维统计特征，由单轮更新分析扩展到跨轮历史行为建模，显著提升了联邦学习的安全性。然而，在动态异构网络环境中，客户端数据普遍呈现 Non-IID 特征，良性客户端更新分布更加复杂，现有方法仍存在以下不足：1)多数方法建立在良性客户端更新满足统一全局一致性的假设之上，当数据异质性增强时，良性客户端之间天然存在较大的更新差异，传统判别标准容易误判正常客户端，导致防御性能下降；2)多数方法主要依据距离、坐标统计量或方向一致性等单一特征识别恶意更新，难以同时刻画方向偏移、更新幅值和高维结构扰动，对复杂模型投毒和隐蔽攻击的适应能力有限；3)部分方法采用硬剔除策略直接舍弃可疑客户端更新，未充分利用客户端可信程度的连续差异，也缺乏对客户端历史行为的持续建模，在恶意比例较高或数据异质性较强时容易造成有效训练信息损失，影响全局模型性能。

2 预备知识

本节介绍本文方法设计所涉及的基础知识，包括联邦学习基本框架、客户端数据分布特点、投毒攻击模型以及后续可信评估中使用的方向一致性、主子空间残差和可信加权聚合等关键技术。其中，客户端数据分布用于分析 IID 与 Non-IID 场景下良性更新统计特性的差异，方向一致性和主子空间残差用于构建客户端可信评估指标，可信加权聚合用于实现最终鲁棒聚合，为第 3 节方法设计奠定基础。

2.1 联邦学习

设联邦学习中总共有 K 个客户端，第 i 个客户端拥有本地数据集 D_i，样本数为 n_i，总样本数为 $n=\sum_{i=1}^{K} n_i$。联邦学习的全局优化目标可以表示为：

$$F(w)=\sum_{i=1}^{K}\frac{n_i}{n}F_i(w)\ , \tag{1}$$

其中，w 表示模型参数，$F_i(w)$ 表示第 i 个客户端的局部经验风险。

$$F_i(w)=\frac{1}{n_i}\sum_{(x_s,y_s)\in D_i}\ell(w;x_s,y_s), \tag{2}$$

其中，$\ell(\cdot)$表示单样本损失函数，x_s 和 y_s 分别表示输入样本及其标签。式(1)(2)表明，联邦学习本质上仍是全局经验风险最小化问题，只是优化过程被拆分到不同客户端本地完成。

联邦学习通过多客户端协同优化完成全局模型训练，服务器仅能获取客户端上传的模型更新，而无法直接访问本地训练数据。因此，服务器只能依据客户端更新判断其可信程度并完成鲁棒聚合，这也是本文投毒防御方法设计的基本出发点。

2.2 客户端数据分布

在理想情况下，如果所有客户端样本均由同一总体分布独立采样得到，且样本类别比例和特征分布大体一致，则可认为客户端数据满足 IID，其任意 2 个客户端 i 和 j 的数据分布满足：

$$P_i(x,y)=P_j(x,y),\quad \forall i,j\ 。 \tag{3}$$

在 IID 条件下，不同良性客户端具有相近的局部目标函数，其模型更新通常围绕相似的优化方向波动。因此，良性更新在方向、幅值和结构上具有较强一致性，为基于全局一致性识别恶意更新提供了基础。

现实联邦学习中更常见的是 Non-IID 场景，即不同客户端的数据在类别比例、样本规模、特征分布或标签语义上存在差异，其数据分布满足：

$$P_i(x,y)\neq P_j(x,y),\quad \exists i\neq j\ 。 \tag{4}$$

Non-IID 现象广泛存在于移动终端、智慧医疗、工业互联网和跨地域业务节点等动态异构网络环境中。例如，不同医院收治患者群体存在差异，不同移动终端保存的图片类别各不相同，不同工业设备采集的数据也具有明显的工况差异。这些差异使得不同客户端优化目标存在一定偏移，即使所有客户端均为良性，其上传模型更新在方向、幅值和结构上也可能表现出较大的自然差异，而不再满足统一的全局一致性。

由此可见，客户端数据分布直接决定了良性更新的统计特性。在 IID 场景下，良性更新通常具有较强的全局一致性；而随着数据异质性的增强，良性更新之间的差异逐渐增大，仅依赖全局统计特征容易将正常异质性误判为恶意行为。因此，在设计联邦学习投毒防御方法时，需要充分考虑不同数据分布条件下客户端更新统计特性的变化。

2.3 投毒攻击

数据投毒攻击的核心是对客户端的本地训练数据进行修改。攻击者可以篡改样本标签、插入带触发器的样本，或者通过重采样和数据污染改变本地数据分布，使客户端在正常训练过程中学习到错误模式。设第 i 个客户端的良性数据集为 D_i，攻击者插入的恶意数据集合为 D_i^{mal}，则被污染后的本地数据集可表示为：

$$D_i'=D_i\cup D_i^{\mathrm{mal}}。 \tag{5}$$

模型投毒攻击不一定修改客户端本地数据，而是直接操纵客户端上传的模型参数或梯度更新。设第 i 个客户端正常更新为 Δ_i^t，攻击者操纵后的上传更新为 $\tilde{\Delta}_i^t$，则模型投毒过程可表示为：

$$\tilde{\Delta}_i^t=\mathcal{A}\left(\Delta_i^t;\theta\right)\ , \tag{6}$$

其中，$\mathcal{A}(\cdot)$ 表示攻击映射，θ 表示攻击参数。攻击者可以对正常更新进行放大、反向、稀疏化、添加后门方向或进行模型替换，使上传更新在服务器聚合时对全局模型产生更直接的影响。

相较于数据投毒，模型投毒通常具有更强攻击能力。攻击者可以直接围绕聚合规则构造恶意更新，从而绕过部分基于距离或统计量的防御方法。例如，针对 Krum、Median 等鲁棒聚合器，攻击者可以设计在距离或坐标统计上不明显异常，但会误导全局优化方向的更新[18-21]。因此，模型投毒防御不能仅依赖单一幅值或距离指标，还需要综合考虑方向、结构和历史行为等多方面特征。

2.4 客户端更新方向一致性

在联邦学习中，客户端上传的模型更新不仅包含幅值大小信息，也包含优化方向信息。对于一个正常参与训练的客户端而言，其本地更新方向通常应当与全局任务的下降方向保持一定一致性。尤其在 IID 场景下，不同良性客户端面对相似的数据分布和优化目标，其模型更新往往会围绕某一主流方向集中分布。因此，方向一致性可以作为判断客户端更新是否异常的重要依据[15-18]。

余弦相似度是衡量向量方向一致性的常用指标。本文先将第 i 个客户端第 t 轮增量 Δ_i^t 归一化为 $\boldsymbol{u}_i^t$，并由当前轮客户端更新估计共识方向 $\boldsymbol{c}^t$。客户端更新方向与共识方向之间的余弦一致性可定义为：

$$q_i^t=\frac{\langle \boldsymbol{u}_i^t,\boldsymbol{c}^t\rangle}{\|\boldsymbol{u}_i^t\|_2\|\boldsymbol{c}^t\|_2}\ , \tag{7}$$

其中，$\langle\cdot,\cdot\rangle$ 表示向量内积，$\|\cdot\|_2$ 表示二范数。当 q_i^t 越接近 1 时，说明客户端更新方向与共识方向越一致；当其接近 0 或小于 0 时，说明该更新与主流优化方向相关性较弱甚至存在冲突。

在 IID 场景下，良性客户端更新通常具有较强的方向一致性。如果某个客户端的更新方向明显偏离共识方向，则该客户端可能存在标签污染、后门注入或模型更新操纵等异常行为。在 Non-IID 场景下，正常客户端之间的数据分布可能存在显著差异，其更新方向不一定围绕单一全局中心集中，需要结合局部结构进行分析。

2.5 客户端更新主子空间残差分析

在高维模型参数空间中，恶意更新并不一定只表现为方向偏离或幅值异常。攻击者可以通过控制更新幅度，使恶意更新在范数上接近正常客户端，也可通过模拟主流更新方向提高方向相似性。然而，即使恶意更新在方向和幅值上具有一定伪装性，其仍可能在高维空间中携带偏离正常更新结构的异常分量。为识别这类结构性异常，本文引入主子空间残差分析[39]。

奇异值分解是提取高维数据主要变化方向的常用方法。设当前轮 m 个客户端的中心化更新向量按行堆叠形成矩阵 $\boldsymbol{Z}^t\in\mathbb{R}^{m\times d}$，其中 d 为模型更新向量维度，则矩阵 $\boldsymbol{Z}^t$ 的奇异值分解可表示为：

$$\boldsymbol{Z}^t=\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}}, \tag{8}$$

其中，$\boldsymbol{U}$ 和 $\boldsymbol{V}$ 分别表示左、右奇异向量矩阵，$\boldsymbol{\Sigma}$ 为奇异值对角矩阵。矩阵 $\boldsymbol{V}$ 中对应最大奇异值的右奇异向量 $\boldsymbol{v}_1^t$ 可以视为客户端更新的主方向，用于刻画当前更新集合中的主要变化趋势。

设第 i 个客户端的中心化更新向量为 $\boldsymbol{z}_i^t$，则其相对于主方向 $\boldsymbol{v}_1^t$ 的投影残差可表示为：

$$e_i^t=\left\|\boldsymbol{z}_i^t-\left((\boldsymbol{z}_i^t)^{\mathrm{T}}\boldsymbol{v}_1^t\right)\boldsymbol{v}_1^t\right\|_2。 \tag{9}$$

设有一组标量观测值 $X=\{x_1,x_2,\cdots,x_m\}$，其中位数记为 $\mathrm{med}(X)$，则中位数绝对偏差可定义为：

$$\mathrm{MAD}(X)=\mathrm{med}(|x_i-\mathrm{med}(X)|)。 \tag{10}$$

为了与其他异常指标统一尺度，本文进一步将投影残差进行鲁棒标准化，得到谱残差惩罚：

$$S_i^t=\max\left(0,\frac{e_i^t-\mathrm{med}(e^t)}{\mathrm{MAD}(e^t)+\varepsilon}\right)。 \tag{11}$$

当 S_i^t 较小时，说明该客户端更新能够较好地被主要变化方向解释，更可能属于正常更新结构；当 S_i^t 较大时，说明该客户端更新在主方向之外存在较多额外分量，可能对应恶意扰动或异常训练行为。

主子空间残差分析的优势在于，它不仅关注客户端更新是否远离中心，还关注其是否偏离正常更新形成的主要结构。在本文中，该思想用于构造谱残差惩罚项，识别那些方向和幅值相对隐蔽但在高维结构上异常的客户端更新。此外，该思想进一步被局部化，即在每个客户端的近邻集合内提取局部主方向，并计算其相对于局部主子空间的结构残差，从而适应 Non-IID 场景下良性更新多簇分布的特点。

2.6 基于客户端可信度的加权聚合

传统鲁棒聚合方法常采用硬剔除策略，然而这类方法在恶意更新明显离群时具有一定效果，但也可能误删包含有效训练信息的边缘良性客户端。尤其在 Non-IID 场景下，良性客户端更新受数据异质性的影响本身存在较大差异，硬剔除可能进一步加剧有效信息损失，影响全局模型的收敛性能。本文采用基于客户端可信度的加权聚合策略，该策略首先根据客户端更新的方向、幅值、结构及历史行为等特征计算可信评分，再将可信评分映射为连续聚合权重，使可信度较高的客户端获得更大的聚合贡献，而可信度较低的客户端贡献被平滑抑制，而非直接剔除。

设第 i 个客户端在第 t 轮的异常分数为 r_i^t，且分数越大表示越可疑。若已筛选得到低风险集合 C_*^t，则可通过 Softmax 形式将异常分数映射为聚合权重：

$$\alpha_i^t=\frac{\exp(-\tau r_i^t)}{\sum\limits_{j\in C_*^t}\exp\left(-\tau r_j^t\right)},\quad i\in C_*^t, \tag{12}$$

其中，C_*^t 表示第 t 轮保留的低风险客户端集合，τ 为温度系数。τ 越大，权重分布越集中，低风险客户端获得更高权重；τ 越小，权重分布越平滑，各客户端之间的权重差异相对减小。

若使用可信分数 s_i^t 表示客户端可信程度，分数越大表示客户端越可信，此时权重可改写为：

$$\alpha_i^t=\frac{\exp\left(\tau s_i^t\right)}{\sum\limits_{j\in C_*^t}\exp\left(\tau s_j^t\right)},\quad i\in C_*^t。 \tag{13}$$

最终聚合更新表示为：

$$\boldsymbol{\Delta}^t=\sum_{i\in C_*^t}\alpha_i^t\boldsymbol{\Delta}_i^t。 \tag{14}$$

3 面向数据异质性的联邦学习投毒防御框架

客户端数据分布直接影响良性更新的统计特性，进而影响投毒防御方法的设计思路。为构建适用于

不同数据分布条件的统一防御框架，本文以 IID 和 Non-IID 两类典型场景为研究对象。首先，以 IID 场景作为基础情形，研究良性客户端更新具有较强全局一致性时的恶意更新识别方法；进一步针对 Non-IID 场景中数据异质性导致的更新分散问题，研究适用于复杂数据分布条件的可信评估与聚合策略，实现防御方法由 IID 场景向 Non-IID 场景的自然扩展。

基于上述思路，本文构建由 ConTrust 和 HistTrust 组成的联邦学习投毒防御框架。其中，ConTrust 作为基础防御方法，主要利用客户端更新的全局一致性特征评估更新可信度，适用于 IID 场景；HistTrust 则在 ConTrust 基础上进一步进行历史行为建模，持续刻画客户端可信度变化，以适应 Non-IID 场景。2 种方法既分别针对不同数据分布特点进行设计，又形成由基础防御到动态防御的统一框架，为不同数据异质程度下的联邦学习投毒防御提供支持。

3.1 威胁模型

1）攻击者目标。攻击者希望持续植入恶意更新来改变全局模型的优化方向，同时保证恶意更新不能与良性更新差异过大，进而绕过服务器端的鲁棒检测。

2）攻击者能力。攻击者能够控制联邦学习中的部分客户端，并对这些客户端的本地训练过程进行完全操纵。攻击者可以在受控客户端的本地数据中插入带有触发器的样本，修改样本标签，或者通过重新采样、污染训练集等方式构造恶意数据分布。攻击者可以直接修改本地优化目标、放大或缩小更新幅值、改变梯度方向，甚至在上传前对本地模型参数进行二次处理，以提升恶意更新对服务器聚合结果的影响。攻击者具备持续投毒能力，而不仅仅局限于单轮次攻击。

此外，攻击者无法知晓服务器端的聚合算法，无法获取其他良性客户端的原始训练数据，且其控制的恶意客户端数量不超过参与聚合客户端数量的半数。

3.2 基于多维一致性评分的防御方法

设第 t 轮服务器端全局模型参数为 $\boldsymbol{w}^t$，参与本轮训练的客户端集合为 C^t，且 $|C^t|=m$。第 i 个客户端完成本地训练后向服务器上传本地模型参数 $\boldsymbol{w}_i^t$。ConTrust 不直接在原始模型参数上进行比较，而是首先计算客户端相对于全局模型的参数增量，即：

$$\boldsymbol{\Delta}_i^t = vec(\boldsymbol{w}_i^t - \boldsymbol{w}^t), \tag{15}$$

其中，$vec(\cdot)$ 表示参数展平算子，用于将各层浮点型参数拼接为统一的一维向量。

在获得所有客户端的增量向量后，ConTrust 首先构建归一化更新矩阵，利用其估计当前轮次的主流优化方向；随后，从方向偏离、幅值偏离和主子空间残差 3 个维度计算每个客户端的异常惩罚项；再将 3 类惩罚加权求和形成混合异常分数；最后根据分数排序得到低风险客户端集合，并在该集合内通过 Softmax 规则分配连续聚合权重。全局模型更新公式可写为：

$$\boldsymbol{w}^{t+1} = \boldsymbol{w}^t + \eta_s \sum_{i \in C^t} \alpha_i^t \boldsymbol{\Delta}_i^t, \tag{16}$$

其中，η_s 为服务器端聚合学习率，$\alpha_i^t \geqslant 0$ 为第 i 个客户端在第 t 轮的聚合权重。与 FedAvg 的样本量加权不同，ConTrust 中的 α_i^t 由客户端风险分数自适应决定，因此低风险客户端将获得更高聚合贡献，而高风险客户端则被显著降权甚至直接截断。

ConTrust 由更新表征、异常评分和加权聚合 3 个核心模块组成，逐层递进，实现对恶意客户端更新的识别与抑制。首先，在更新表征阶段，服务器接收各客户端上传的本地模型参数后，将其转换为相对于当前全局模型的参数增量，以消除不同轮次全局模型基准差异带来的影响；同时，对参数增量进行归一化处理，从而降低模型规模、更新尺度以及客户端数据分布差异对后续度量的干扰。随后，在异常评分阶段，从共识方向偏离、更新幅值偏离和主子空间残差 3 个维度评估客户端更新的异常程度，并融合各维度信息形成混合异常分数，以全面刻画客户端更新的可信程度。最后，在加权聚合阶段，服务器依据异常分数筛选出低风险客户端集合，并根据风险水平分配连续聚合权重，使可信更新获得更高贡献，而潜在恶意更新被平滑抑制。整体来看，更新表征为异常评分提供统一的特征输入，异常评分量化不同客户端的风险水平，加权聚合则将风险评估结果转化为鲁棒聚合权重，共同构成完整的防御流程。

1）更新向量表征与共识方向建模

本模块旨在刻画良性客户端整体优化趋势的共识方向，为后续异常评分提供统一参考。由于模型更新同时包含方向信息和幅值信息，直接比较原始更新容易受到更新尺度差异的影响。因此，ConTrust 首先对客户端模型更新进行归一化，将方向信息与幅值信息解耦，使方向一致性分析不受更新范数的干扰。

设第 i 个客户端在第 t 轮上传的模型更新为 $\boldsymbol{\Delta}_i^t$，其归一化更新向量定义为：

$$\boldsymbol{u}_i^t = \frac{\boldsymbol{\Delta}_i^t}{\|\boldsymbol{\Delta}_i^t\|_2 + \varepsilon}, \tag{17}$$

其中，ε 为避免除零错误的极小常数。归一化后，各客户端更新保留优化方向信息，而更新幅值由后续模块独立分析，从而避免大范数更新对共识方向估计产生过度影响。

在 IID 场景下，良性客户端的更新方向通常围绕共同优化方向分布。因此，ConTrust 使用所有归一化更新的均值来估计当前轮次的共识方向 $\boldsymbol{c}^t$：

$$\bar{\boldsymbol{u}}^t = \frac{1}{m}\sum_{i\in C^t}\boldsymbol{u}_i^t, \quad \boldsymbol{c}^t = \frac{\bar{\boldsymbol{u}}^t}{\|\bar{\boldsymbol{u}}^t\|_2 + \varepsilon}。 \tag{18}$$

在得到共识方向后，利用余弦相似度衡量各客户端更新与共识方向之间的一致性：

$$q_i^t = \left\langle \boldsymbol{u}_i^t, \boldsymbol{c}^t \right\rangle。 \tag{19}$$

若客户端更新方向与共识方向高度一致，则 q_i^t 接近 1；若更新方向明显偏离甚至相反，则该值会显著下降。考虑到均值和标准差容易受到异常客户端干扰，本文采用基于中位数 $\mathrm{med}(\cdot)$ 和中位数绝对偏差 $\mathrm{MAD}(\cdot)$ 的鲁棒标准化方式[12,14]计算方向偏离程度：

$$\phi_{(x_i)} = \max\left(0, \frac{\mathrm{med}(x) - x_i}{\mathrm{MAD}(x) + \varepsilon}\right)。 \tag{20}$$

此时第 i 个客户端的方向惩罚被定义为：

$$D_i^t = \phi_{(q_i^t)}。 \tag{21}$$

方向惩罚反映客户端更新偏离共识方向的程度。客户端更新与共识方向越一致，其方向惩罚越小；偏离程度越大，则对应惩罚越高。

2）幅值偏离与主子空间残差评估

方向一致性能够反映客户端更新的整体优化趋势，但难以识别方向相近而更新幅值异常或内部结构异常的恶意更新。因此，ConTrust 进一步从更新幅值和主子空间结构 2 个角度补充异常评估，提高复杂攻击场景下的识别能力。

首先，针对更新幅值异常问题，设第 i 个客户端更新范数为 $n_i^t = \|\boldsymbol{\Delta}_i^t\|_2$。为减弱不同轮次更新尺度差异的影响，首先对更新范数进行对数变换，再采用鲁棒标准化计算幅值偏离惩罚：

$$A_i^{(t)} = \frac{\left|\log\left(n_i^t + \varepsilon\right) - \mathrm{med}_j \log\left(n_j^t + \varepsilon\right)\right|}{\mathrm{MAD}_j\left(\log\left(n_j^t + \varepsilon\right)\right) + \varepsilon}。 \tag{22}$$

幅值偏离惩罚用于刻画客户端更新规模偏离整体统计分布的程度。更新范数越偏离正常客户端的集中分布，其对应的幅值偏离惩罚越大，从而能够有效识别更新尺度异常的客户端。

其次，为识别高维更新中的结构性异常，ConTrust 利用主子空间残差衡量客户端更新是否偏离良性客户端形成的主要变化子空间。设所有归一化更新的均值为 $\bar{\boldsymbol{u}}^t$，则第 i 个客户端的中心化向量为：

$$\boldsymbol{z}_i^t = \boldsymbol{u}_i^t - \bar{\boldsymbol{u}}^t。 \tag{23}$$

将所有中心化向量按行堆叠构成矩阵 $\boldsymbol{Z}^t$，并进行奇异值分解：

$$\boldsymbol{Z}^t = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}}。 \tag{24}$$

取 $\boldsymbol{V}$ 中对应最大奇异值的右奇异向量 $\boldsymbol{v}_1^t$ 作为主方向，即当前轮次中客户端更新的主要变化子空间方向。当良性客户端占多数且满足 IID 分布时，归一化更新的中心化矩阵通常具有较强低秩特征，其主要能量集中在主方向上。反之，恶意客户端为了投毒，会在主方向之外附加较多正交分量。基于此，得到第 i 个客户端在主方向之外的投影残差：

$$e_i^t = \left\| \left(\boldsymbol{I} - \boldsymbol{v}_1^t (\boldsymbol{v}_1^t)^{\mathrm{T}}\right) \boldsymbol{z}_i^t \right\|_2。 \tag{25}$$

进一步得到谱残差惩罚：

$$S_i^t = \max\left(0, \frac{e_i^t - \mathrm{med}(e^t)}{\mathrm{MAD}(e^t) + \varepsilon}\right)。 \tag{26}$$

从几何意义上看，S_i^t 描述的是客户端更新偏离主子空间的程度。如果某个客户端更新在平均方向上保持一致性，但在高维空间中携带额外异常扰动，仍能够在谱残差指标上被有效识别。

3）截断式平滑加权聚合

在获得方向惩罚 D_i^t、幅值惩罚 A_i^t 和谱残差惩罚 S_i^t 之后，ConTrust 通过线性加权方式构建客户端的混合异常分数：

$$r_i^t = \lambda_D D_i^t + \lambda_A A_i^t + \lambda_S S_i^t, \tag{27}$$

其中，λ_D、λ_A 和 λ_S 分别为 3 个模块的权重系数。默认配置分别设为 2.4、1.2 和 1.4，以突出方向一致性指标的重要性，同时兼顾更新幅值与谱残差异常。

若直接基于 r_i^t 对所有客户端进行软加权，会导致极少数高风险客户端虽然权重很小，但在攻击强度较大时仍可能对聚合结果产生影响。为此，ConTrust 采用“先截断、再平滑”的策略。设客户端总数为 m，保留比例为 ρ，保留客户端数量为 $k = \lceil \rho m \rceil$，则先选择分数最小的前 k 个客户端：

$$C_*^t = \left\{ i \in C^t \mid r_i^t \leqslant r_{(k)}^t \right\}, \quad k = \lceil \rho m \rceil, \tag{28}$$

其中，$r_{(k)}^t$ 表示当前轮所有客户端异常分数中的第 k 小值。根据默认设置，$\rho = 0.5$，即每轮优先保留风险最低的一半客户端。该设置隐含了一个符合鲁棒聚合常识的假设：良性客户端应占多数。因此分数最低的一半客户端中大概率包含大部分良性更新。

在保留集合 C_*^t 内，继续用温度系数 τ 的 Softmax 规则分配连续权重：

$$\alpha_i^t=\begin{cases}\dfrac{\exp\left(-\tau r_i^t\right)}{\sum\limits_{j\in C_*^t}\exp\left(-\tau r_j^t\right)}, & i\in C_*^t\\ 0, & i\notin C_*^t\end{cases}, \tag{29}$$

其中，默认温度参数为 $\tau=2.5$。当 τ 较大时，较低风险客户端会获得更集中的权重；当 τ 较小时，权重分布更均匀。为避免权重过度集中，对聚合权重设置最小权重下界，并重新归一化。因此，最终聚合结果既保留了低风险客户端贡献的连续差异，同时避免了恶意客户端通过尾部权重渗透全局模型。

ConTrust 具体防御过程如算法 1 所示。

算法 1. 基于多维一致性评分的防御方法。

输入：当前全局模型 $\boldsymbol{w}^t$，参与聚合的客户端模型集合 $\{\boldsymbol{w}^t\}_{i\in C^t}$，服务器学习率 η_s，客户端保留比例 ρ，权重系数 $\lambda_D,\lambda_A,\lambda_S$，温度系数 τ；

输出：下一轮全局模型 $\boldsymbol{w}^{t+1}$。

① 初始化客户端数量 $m\leftarrow|C^t|$；

② for each 客户端 $i\in C^t$ do

③ 计算客户端参数增量 $\boldsymbol{\Delta}_i^t$（式 15）；

④ 计算客户端更新范数 n_i^t，归一化更新向量 $\boldsymbol{u}_i^t$；

⑤ end for

⑥ 计算所有归一化更新计算共识方向 $\boldsymbol{c}^t$（式 18）；

⑦ for each 客户端 $i\in C^t$ do

⑧ 计算多维惩罚项 D_i^t，A_i^t，S_i^t（式（21），（22），（26））；

⑨ 计算混合异常分数 r_i^t（式（27））；

⑩ end for

⑪ 根据 ρ 构造低风险集合 C_*^t（式（28））；

⑫ for $i=1$ to m do

⑬ 使用基于 τ 的 Softmax 计算权重 α_i^t（式（29））；

⑭ end for

⑮ 施加最小权重下界 $\alpha_{\min}$，并重新归一化；

⑯ 更新全局模型 $\boldsymbol{w}^{t+1}$（式（16））；

⑰ return $\boldsymbol{w}^{t+1}$。

3.3 基于历史行为建模的防御方法

针对 Non-IID 场景下数据异质性导致全局一致性假设失效的问题，HistTrust 按照“局部模式感知—历史行为信誉建模—动态可信聚合”的逐层处理流程，通过融合当前轮局部行为与跨轮历史行为，实现客户端可信度的动态评估与鲁棒聚合。首先，在局部模式感知即时信任评估阶段，根据当前轮各客户端更新之间的局部近邻关系，为每个客户端构建更符合其数据分布特征的局部参考，并从局部方向、局部幅值和局部结构残差 3 个维度评估客户端当前轮行为，得到即时信任分；其次，在历史行为信誉累积阶段，对客户端多轮即时信任分进行指数滑动方式累积，构建历史信誉，并进一步刻画信任波动程度；最后，在自举式动态可信聚合阶段，将上一轮历史行为得分作为当前轮可信评估的先验信息，与即时信任分联合生成客户端动态可信度，并据此筛选可信客户端集合和分配聚合权重。

与 ConTrust 基于单轮全局一致性的可信评估不同，HistTrust 将客户端可信度建模扩展为局部行为与历史行为的联合评估。其中，判别基线由全局单中心转变为局部近邻模式，可信依据由单轮静态评分扩展为即时信任与历史信誉的联合建模，从而实现面向 Non-IID 场景的动态可信聚合。

1）局部模式感知即时信任评估

在 Non-IID 场景下，局部模式感知即时信任评估旨在判断客户端当前轮更新是否可信。由于良性客户端更新受数据异质性的影响通常呈现多个局部聚类，统一采用全局参考容易将正常更新误判为异常。因此，HistTrust 不再以全局更新作为唯一判别基线，而是为每个客户端构建局部近邻参考，并基于局部模式评估客户端更新的方向一致性、幅值偏离和结构一致性。与 ConTrust 保持一致，HistTrust 仍采用方向惩罚 D_i^t、幅值惩罚 A_i^t 和结构残差惩罚 S_i^t 构建当前轮风险分值 r_i^t，其计算方式与式（27）一致。

首先，同样对客户端参数增量进行归一化，解耦方向信息与幅值信息，得到 $l_i^t=\log\left(\|\boldsymbol{\Delta}_i^t\|_2+\varepsilon\right)$，此时客户端 i 与客户端 j 在第 t 轮的联合距离为：

$$d_{ij}^t=\lambda_{\cos}\left(1-\langle\boldsymbol{u}_i^t,\boldsymbol{u}_j^t\rangle\right)+\lambda_n\left|l_i^t-l_j^t\right|, \tag{30}$$

其中，$1-\langle\boldsymbol{u}_i^t,\boldsymbol{u}_j^t\rangle$ 表述客户端更新方向差异，$\left|l_i^t-l_j^t\right|$ 描述更新幅值差异，$\lambda_{\cos}$ 与 λ_n 为平衡系数，分别控制 2 部分距离的相对贡献。与 ConTrust 基于全体客户端共享同一全局方向不同，HistTrust 使用联合距离为每个客户端单独构建局部近邻集合。设 k 为近邻数量，则客户端 i 的局部邻域可表示为：

$$\mathcal{N}_i^t=TopK_{\min}\left(\{d_{ij}^t|j\in C_t,j\neq i\},k\right)。 \tag{31}$$

随后计算客户端 i 所属局部模式的方向中心：

$$\boldsymbol{c}_i^t=\frac{\sum\limits_{j\in\mathcal{N}_i^t\cup\{i\}}\boldsymbol{u}_j^t}{\left\|\sum\limits_{j\in\mathcal{N}_i^t\cup\{i\}}\boldsymbol{u}_j^t\right\|_2+\varepsilon}。 \tag{32}$$

当客户端更新方向与其局部模式中心保持较高

一致性时,说明该更新与所在局部模式具有较好的优化一致性,更可能属于正常客户端;反之,若其更新方向明显偏离局部模式中心,则表明该客户端可能存在异常更新行为。

设 $q_i^t=\langle \boldsymbol{u}_i^t,\boldsymbol{c}_i^t\rangle$,并令 $\Phi_i^t=\left\{\left\langle \boldsymbol{u}_j^t,\boldsymbol{c}_i^t\right\rangle \mid j\in \mathcal{N}_i^t\cup\{i\}\right\}$,则局部方向惩罚定义为:

$$D_i^t=\max\left\{0,\frac{\operatorname{med}\left(\Phi_i^t\right)-q_i^t}{\operatorname{MAD}\left(\Phi_i^t\right)+\varepsilon}\right\},\quad q_i^t=\left\langle \boldsymbol{u}_i^t,\boldsymbol{c}_i^t\right\rangle 。 \tag{33}$$

2)历史行为信誉累积

局部模式感知即时信任评估主要依据当前轮客户端更新特征判断其可信程度,反映客户端的即时行为。然而,仅依赖单轮信息进行聚合仍存在一定局限。一方面,受数据异质性和训练随机性的影响,良性客户端更新可能在个别轮次出现正常波动,仅依据当前轮结果容易产生误判;另一方面,恶意客户端可通过降低攻击强度、间歇参与或持续缓慢偏移等策略规避单轮检测。为提高客户端可信评估的稳定性,HistTrust 进一步对客户端跨轮行为进行建模,通过历史信誉与波动记忆共同描述客户端长期可信程度。

设 h_i^t 表示客户端 i 截至第 t 轮的历史信誉。若客户端 i 参与第 t 轮训练,则依据即时信任分 s_i^t 对历史信誉执行指数平滑更新;否则保持历史信誉不变。历史信誉更新公式为:

$$h_i^t=\mu h_i^{t-1}+(1-\mu)s_i^t, \tag{34}$$

其中,$\mu\in[0,1)$ 为历史记忆系数,用于控制历史信誉与当前即时信任之间的权衡。μ 越大,历史信誉保留程度越高,客户端长期行为对信誉更新的影响越显著,有利于增强对长期潜伏型恶意客户端的抑制能力;μ 越小时,历史信誉更新更加依赖当前即时信任,能够更快反映客户端近期行为变化,提高信誉评估对动态训练过程的响应能力。

然而,仅依据历史信誉仍不足以完整描述客户端长期行为,因为 2 个客户端可能具有相近的平均信誉,但其中一个客户端跨轮表现平稳,另一个客户端则频繁在可信和可疑之间波动。对服务器而言,后者更值得警惕,因为这类行为模式更接近潜伏与攻击交替执行的恶意策略。因此,HistTrust 进一步引入信任波动记忆 $\boldsymbol{v}_i^t$,用于刻画客户端即时信任相对于既有历史信誉的偏离程度:

$$\boldsymbol{v}_i^t=\mu_v\boldsymbol{v}_i^{t-1}+(1-\mu_v)\left|s_i^t-h_i^{t-1}\right| 。 \tag{35}$$

若客户端跨轮行为具有较好一致性时,则其当前即时信任与历史信誉之间的差异通常较小,$\boldsymbol{v}_i^t$ 将保持较低水平;反之,若客户端即时信任相对于历史信誉持续产生较大偏离,则波动记忆将逐步增大。综合历史信誉与波动记忆,客户端历史行为得分可定义为:

$$\boldsymbol{\xi}_i^t=h_i^t-\lambda_v\boldsymbol{v}_i^t, \tag{36}$$

其中,λ_v 为波动惩罚系数,用于控制波动记忆对历史行为得分的影响。历史行为得分同时融合了客户端的长期可信程度和跨轮行为稳定性,为后续自举式动态可信聚合提供可靠的历史先验。

3)自举式动态可信聚合

在得到当前轮即时信任 s_i^t 与历史行为得分 $\boldsymbol{\xi}_i^t$ 之后,如何将过去的长期判断与当前轮的即时评估共同转化为可执行的聚合权重成为新的难题。为避免使用当前轮聚合结果之后才更新得到的历史状态参与本轮决策,HistTrust 在第 t 轮权重分配时使用上一轮历史行为得分 $\boldsymbol{\xi}_i^{t-1}$ 作为先验,再与当前轮即时信任 s_i^t 联合形成最终自举可信度。

客户端 i 在第 t 轮的自举可信度 b_i^t 可表示为:

$$b_i^t=\beta\boldsymbol{\xi}_i^{t-1}+(1-\beta)s_i^t, \tag{37}$$

其中,$\beta\in[0,1]$ 为先验权重系数。当 β 较大时,更看重历史信誉,适合对抗长期潜伏型恶意客户端;当 β 较小时,系统更强调当前轮表现,适合快速适应局部数据分布发生明显变化的情况。

与 ConTrust“先截断、再平滑”聚合思想一致,HistTrust 仍在权重分配前先保留可信客户端子集。不同之处在于,ConTrust 依据单轮混合异常分数排序,而 HistTrust 依据自举可信度排序。同样地,设 ρ 为保留比例,保留客户端数量为 $k_b=\lceil\rho m\rceil$,则保留集合 C_*^t 可表示为:

$$C_*^t=TopK_{\max}\left(\{b_i^t|i\in C^{(t)}\},k_b\right),\quad k_b=\lceil\rho m\rceil 。 \tag{38}$$

在保留集合 C_*^t 内,再利用温度系数 τ 执行平滑加权,得到客户端最终聚合权重:

$$\alpha_i^t=\frac{\exp\left(\tau b_i^t\right)}{\sum\limits_{j\in C_*^t}\exp\left(\tau b_j^t\right)},\quad i\in C_*^t 。 \tag{39}$$

最终全局模型更新公式与 ConTrust 式(16)保持一致,服务器端聚合学习率仍然记为 η_{s}:

$$\boldsymbol{w}^{t+1}=\boldsymbol{w}^t+\eta_{\mathrm{s}}\sum_{i\in C_*^t}\alpha_i^t\boldsymbol{\Delta}_i^t 。 \tag{40}$$

自举式动态可信聚合通过将历史信誉纳入后续聚合决策,实现客户端可信度的跨轮迭代更新。随着训练不断进行,长期稳定的良性客户端将逐步积累较高的可信先验,而行为异常或跨轮波动较大的

客户端则持续受到抑制，从而进一步提高数据异质场景下恶意更新识别的准确性和鲁棒聚合的稳定性。HistTrust 具体防御过程如算法 2 所示。

算法 2. 基于历史行为建模的防御方法。

输入：当前全局模型 $\boldsymbol{w}^t$，参与聚合的客户端模型集合 $\{\boldsymbol{w}_i^t\}_{i\in C^t}$，上一轮历史信誉 $\{h_i^{t-1}\}$，上一轮波动记忆 $\{v_i^{t-1}\}$；距离平衡系数 $\lambda_{\cos},\lambda_n$，权重系数 $\lambda_D,\lambda_A,\lambda_S$，历史记忆系数 μ，波动惩罚系数 λ_v，先验权重系数 β，保留比例 ρ，温度系数 τ，服务器学习率 η_s；

输出：下一轮全局模型 $\boldsymbol{w}^{t+1}$，更新后的历史信誉 $\{h_i^t\}$ 与波动记忆 $\{v_i^t\}$。

① 初始化客户端 $\boldsymbol{\Delta}_i^t$、历史信誉和波动记忆；

② for each 客户端 $i\in C^t$ do

③ 计算客户端参数增量 $\boldsymbol{\Delta}_i^t$（式(15)）；

④ end for

⑤ 基于 $\lambda_D,\lambda_A,\lambda_S$ 计算当前轮风险分 r_i^t（式(27)）；

⑥ 计算即时信任 $s_i^t=\exp\left(-r_i^t\right)$；

⑦ for each 客户端 $i\in C^t$ do

⑧ 根据 μ、v_i^{t-1}、h_i^t、λ_v 计算历史行为得分 ξ_i^{t-1}（式(34)(35)(36)）；

⑨ 根据 β、ξ_i^{t-1}、s_i^t 计算自举可信度 b_i^t（式(37)）；

⑩ end for

⑪ 根据 ρ 构造低风险集合 C_*^t（式(38)）；

⑫ 计算聚合权重 α_i^t（式(38)(39)）；

⑬ 更新全局模型 $\boldsymbol{w}^{t+1}$（式(40)）；

⑭ 更新历史信誉 h_i^t 与波动记忆 v_i^t；

⑮ return $\boldsymbol{w}^{t+1}$, $\{h_i^t\}$, $\{v_i^t\}$。

4 实　　验

4.1 实验设置

为验证本文所提防御方法的有效性，本文在统一的实验环境下开展相关实验，具体实验环境配置参数如表 1 所示。

Table 1 Configuration Parameters of Experimental Environment

表 1 实验环境配置参数

参数	配置环境
处理器	Intel® Xeon® Platinum 8352V CPU@2.10 GHz
内存/GB	1 024
Python 版本	3.12
CUDA 版本	12.4
Linux 环境	Ubuntu22.04

1）数据集

本文选取 MNIST[40]、Fashion-MNIST（FMNIST）[41]和 CIFAR-10[42] 三个公开数据集开展实验，以验证所提方法在不同图像分类任务中的防御性能。其中 MNIST 用于手写数字识别任务，包含 60 000 张训练图像和 10 000 张测试图像，图像尺寸为 28×28，均为灰度图像。FMNIST 与 MNIST 具有相同的数据规模和图像尺寸，但将分类对象扩展为 10 类时尚服饰，能更好地评估模型在简单视觉分类任务中的泛化能力；CIFAR-10 包含 10 类自然场景目标，包含 50 000 张训练图像和 10 000 张测试图像，图像尺寸为 32×32，为 RGB 彩色图像。

上述数据集均为联邦学习安全研究中广泛采用的公开基准数据集，具有统一的数据划分标准和成熟的实验设置，能够保证不同方法之间的公平比较。本文关注的是异质数据环境下联邦学习投毒防御算法的有效性，而非特定应用场景的数据特征，因此采用 Dirichlet 分布构建不同程度的 Non-IID 数据划分，以模拟物联网、智慧医疗等真实联邦学习场景中普遍存在的统计异构特征。同时，实验覆盖了灰度图像与自然彩色图像 2 类具有不同复杂度的分类任务，能够从不同数据复杂度和异构程度 2 个维度对所提方法的鲁棒性和泛化能力进行较为全面的验证。

2）攻击方法

为全面评估所提方法对不同类型投毒攻击的防御能力，本文选取标签翻转攻击[23]、DBA[31]、Neurotoxin[33] 和 FCBA[38] 等 4 种代表性攻击方法作为实验对象。其中，标签翻转攻击属于数据投毒方法，通过篡改本地训练样本标签诱导模型学习错误的类别映射，从而降低模型分类性能；DBA 属于分布式后门攻击方法，将完整触发器划分为多个局部子触发器，并分配至不同恶意客户端，以提高攻击的隐蔽性和攻击成功率；Neurotoxin 针对训练过程中变化幅度较小且不易被良性更新覆盖的模型参数实施投毒，从而增强攻击的持久性；FCBA 通过组合触发器信息，使全局模型学习到更完整的后门模式，有效缓解后门在训练过程中的衰减，提升攻击成功率。

3）防御基线

本文选取 FedAvg[3]、Multi-Krum[11]、Trimmed Mean[12]、FL-Defender[43] 和 MinVar[44]、FLTrust[17]、FLDetector[19] 和 FLAIR[9] 等 8 种防御方法作为对比基线。其中，FedAvg 采用加权平均方式聚合客户端模型更新，是联邦学习最经典的聚合算法，但缺乏针对恶意更新的鲁棒性；Multi-Krum 属于基于距离选择的

鲁棒聚合方法,通过选择与多数客户端更新更一致的模型参与聚合,降低异常更新的影响;Trimmed Mean 采用坐标级鲁棒统计策略,在每个参数维度剔除极端值后进行均值聚合,以抑制异常坐标带来的干扰;FL-Defender 基于梯度相似性和信任评分,通过分析客户端最后一层梯度特征,并结合角度相似性、主成分分析及质心偏差评估客户端可信程度,实现更新加权聚合;MinVar 以最小化聚合方差为目标,通过优化客户端聚合权重降低恶意更新对全局模型的影响,而无需直接剔除异常客户端;FLTrust 是基于可信参考更新的信任驱动方法,通过服务器端少量可信数据生成参考更新,并依据客户端更新与参考更新之间的方向一致性分配聚合权重;FLDetector 属于基于历史一致性检测方法,通过比较客户端当前更新与历史预测更新之间的偏差识别异常行为;FLAIR 属于信誉评分方法,依据信誉对客户端更新进行加权聚合。

4)评价指标

本文采用模型准确率(accuracy, ACC)和攻击成功率(attack success rate, ASR)作为评价指标。其中,ACC 表示全局模型在干净测试集上的分类准确率,用于衡量模型主任务性能,ACC 越高表示分类性能越优;ASR 表示带有攻击触发器的测试样本被错误分类为目标类别的比例,用于衡量攻击效果,ASR 越低说明防御方法能更有效抑制攻击。

4.2 实验结果

1)无攻击场景下的主任务性能

图 1 展示了 IID 场景下无攻击时不同防御方法在 3 个数据集上的 ACC。其中,在 MNIST 数据集上,ConTrust 的 ACC 为 96.45%,仅比 Multi-Krum 低 0.13 个百分点,同时略高于 FedAvg (96.38%);同样,在 CIFAR-10 数据集上,ConTrust 的 ACC 达到 75.37%,与

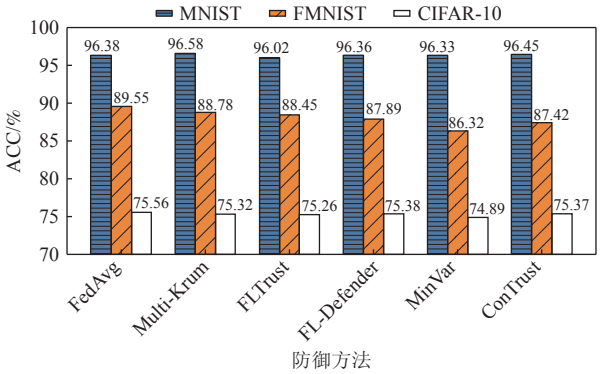


Fig. 1 ACC of different defense methods under IID settings without attacks

图 1 IID 场景下无攻击时不同防御方法的 ACC

FedAvg (75.56%)和 FL-Defender (75.38%)基本一致。

相比之下,在 FMNIST 数据集上,ConTrust 的 ACC 为 87.42%,低于 FedAvg 和 Multi-Krum。这说明在任务难度相对更高、类别纹理差异更小的情况下,ConTrust 的保守加权策略会牺牲部分主任务精度。但与 MinVar 相比,ConTrust 仍高出 1.1 个百分点。综上,ConTrust 在无攻击环境下保持了与现有防御方法相当的 ACC,其主要设计目标并非单纯提升无攻击条件下的最优 ACC,而是在保持主任务性能的基础上,为后续抵御投毒攻击提供更强的鲁棒性。

图 2 展示了强 Non-IID 场景($\alpha = 0.1$)下,无攻击情况下不同防御方法在 3 个数据集上的 ACC,总体来看,HistTrust 能够保持基本稳定的主任务性能。

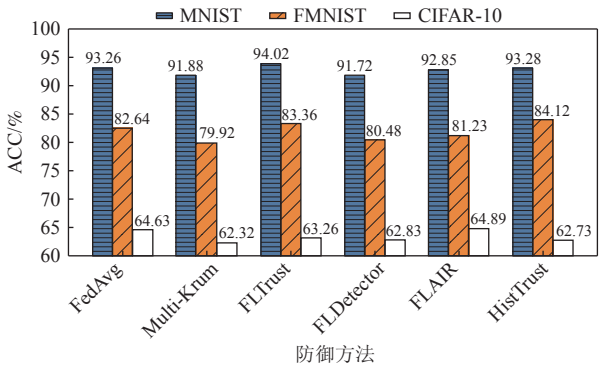


Fig. 2 ACC of different defense methods under Non-IID settings without attacks

图 2 Non-IID 场景下无攻击时不同防御方法的 ACC

具体来看,HistTrust 在 MNIST 上的 ACC 为 93.28%,与 FedAvg (93.26%)基本一致。在 FMNIST 上,HistTrust 取得 84.12% 的最高 ACC,优于 FedAvg、Multi-Krum、FLTrust、FLDetector 和 FLAIR,表明 HistTrust 引入的局部可信评估和历史信誉机制能够有效缓解强 Non-IID 条件下的聚合偏差。在 CIFAR-10 上,HistTrust 的 ACC 为 62.73%,略低于 FedAvg 和 FLAIR。由于 CIFAR-10 具有更高的数据复杂度,且在强 Non-IID 划分下客户端间数据分布差异更加显著,HistTrust 为降低潜在异常更新影响而采用的保守筛选与自适应加权策略,会适度削弱部分正常更新的贡献,从而带来一定的精度损失。

2)默认恶意客户端比例下的总体防御效果

表 2(IID 场景)和表 3(Non-IID 场景)展示了恶意客户端比例为 30% 时,不同防御方法在标签翻转、DBA、Neurotoxin 和 FCBA 四类攻击下的实验结果,其中每个单元格均采用“ACC/ASR”的形式。

如表 2 所示,FedAvg 在各类攻击下均表现出较

Table 2 ACC/ASR of Different Algorithms Under IID Settings with 30% Malicious Clients

表 2 IID场景下30%恶意客户端时不同算法的 ACC/ASR %

数据集	防御方法	标签翻转	DBA	Neurotoxin	FCBA
MNIST	FedAvg	**94.68**/22.56	92.12/30.12	93.56/27.56	88.65/35.23
	Multi-Krum	93.89/9.65	92.89/25.65	92.12/19.23	87.56/28.25
	Trimmed Mean	93.76/8.98	91.65/25.06	92.48/20.66	89.63/25.33
	FL-Defender	94.21/3.98	92.56/12.35	**93.98**/10.64	88.85/14.26
	MinVar	92.26/10.25	91.95/13.36	91.98/11.66	89.63/16.33
	ConTrust	94.55/**3.56**	**93.23**/**11.95**	92.78/**9.45**	**89.67**/**13.89**
FMNIST	FedAvg	77.55/25.73	82.45/33.66	85.56/23.62	80.63/37.54
	Multi-Krum	77.98/7.96	83.23/26.45	85.35/16.26	80.65/30.25
	Trimmed Mean	76.46/8.03	83.54/27.63	84.32/16.45	79.66/29.56
	FL-Defender	78.15/**4.32**	**84.56**/16.14	86.32/**8.84**	81.16/20.12
	MinVar	78.32/8.96	81.99/18.46	86.45/8.24	82.56/22.45
	ConTrust	**78.76**/4.89	84.16/**15.21**	**86.63**/8.32	**83.12**/**20.04**
CIFAR-10	FedAvg	71.46/29.85	65.59/35.66	68.45/30.55	60.03/40.36
	Multi-Krum	72.69/8.34	64.24/27.48	69.12/23.52	61.22/34.21
	Trimmed Mean	71.97/8.23	63.78/28.56	69.09/21.21	61.32/32.15
	FL-Defender	73.08/2.86	66.21/18.46	69.49/15.32	61.32/23.15
	MinVar	72.18/6.23	65.96/17.98	**69.53**/14.56	**62.85**/22.26
	ConTrust	**73.17**/**2.65**	**67.59**/**17.23**	69.11/**14.11**	62.65/**19.95**

注：黑体数值表示最优值。

Table 3 ACC/ASR of Different Algorithms Under Non-IID Settings with 30% Malicious Clients

表 3 Non-IID 场景下 30% 恶意客户端时不同算法的 ACC/ASR %

数据集	防御方法	标签翻转	DBA	Neurotoxin	FCBA
MNIST	FedAvg	89.36/31.74	86.88/45.26	88.05/38.92	81.76/43.58
	Multi-Krum	87.92/18.46	86.24/35.82	87.36/28.14	82.18/39.26
	FLTrust	**90.94**/14.28	88.10/30.45	89.28/24.68	80.72/33.14
	FLDetector	88.54/12.82	87.42/26.84	88.61/18.74	83.36/31.58
	FLAIR	89.74/**9.36**	88.36/24.12	89.04/19.86	**84.08**/29.62
	HistTrust	90.18/10.42	**89.04**/**22.64**	**90.36**/**16.88**	83.18/**26.73**
FMNIST	FedAvg	70.82/39.68	74.56/50.24	77.16/36.28	65.44/48.92
	Multi-Krum	69.28/24.36	73.22/40.84	75.04/30.18	67.86/43.75
	FLTrust	72.56/17.28	75.78/33.46	77.92/23.52	66.68/38.26
	FLDetector	70.46/20.15	74.52/31.64	76.44/20.76	**68.96**/35.84
	FLAIR	71.84/16.12	75.64/27.58	77.28/**15.73**	63.18/34.12
	HistTrust	**73.18**/**13.64**	**76.84**/**23.72**	**79.26**/16.42	68.92/**30.48**
CIFAR-10	FedAvg	56.24/45.36	50.62/58.18	53.28/47.95	47.36/62.74
	Multi-Krum	54.86/29.42	50.08/46.24	52.36/38.16	48.42/52.38
	FLTrust	55.68/23.74	51.74/39.56	53.92/32.84	49.48/47.12
	FLDetector	54.72/25.08	51.18/29.72	53.46/24.82	49.16/45.68
	FLAIR	**58.12**/20.36	53.18/35.24	54.62/25.35	**50.76**/45.64
	HistTrust	57.42/**17.84**	**53.49**/**28.86**	**56.18**/**23.64**	49.86/**32.42**

注：黑体数值表示最优值。

高的 ACC, 难以抵御投毒攻击。以 CIFAR-10 的 FCBA 攻击为例, FedAvg 的 ASR 达到 40.36%, 显著高于其他方法, 表明简单的平均聚合难以阻止恶意更新。相较于 FedAvg, Multi-Krum 和 Trimmed Mean 在多数场景下能够降低 ASR, 但防御效果并不稳定。例如, 在 MNIST 标签翻转攻击下, Multi-Krum 和 Trimmed Mean 分别将 ASR 降低至 9.65% 和 8.98%; 但在 DBA 和 FCBA 等攻击下, 其防御能力明显下降。在 MNIST 数据集上, Multi-Krum 在 DBA 和 FCBA 下的 ASR 分别达到 25.65% 和 28.25%, 说明仅依赖距离选择或坐标统计难以有效区分具有较强隐蔽性的恶意更新。

ConTrust 在多数攻击场景下均取得了最低或接近最低的 ASR, 并保持了较高的 ACC。在 MNIST 数据集上, ConTrust 在 4 类攻击下的 ASR 分别仅为 3.56%、11.95%、9.45% 和 13.89%, 均优于其他方法; 在 CIFAR-10 数据集上, ConTrust 的 ASR 分别为 2.65%、17.23%、14.11% 和 19.95%, 同样取得最优防御效果。上述结果说明, ConTrust 能够有效识别并抑制不同类型的恶意更新, 从而在不同攻击场景下均表现出稳定的鲁棒性。

在 FMNIST 数据集上, ConTrust 的优势主要体现为防御能力与模型性能之间的良好平衡。在标签翻转攻击和 Neurotoxin 攻击下, ConTrust 的 ASR 分别为 4.89% 和 8.32%, 略高于 FL-Defender 和 MinVar 的最优值, 但其 ACC 分别达到 78.76% 和 86.63%, 均为所有方法中的最高值。在 DBA 和 FCBA 攻击下, ConTrust 则同时取得几乎最低的 ASR 和较高的 ACC。由此可见, ConTrust 在 ASR 和 ACC 之间取得了较好的权衡。

进一步统计上述实验的平均结果可以发现, ConTrust 的平均 ACC 达到 81.29%, 平均 ASR 仅为 11.77%。相比 FedAvg, ConTrust 将平均 ASR 从 31.04% 降低至 11.77%, 下降 19.27 个百分点; 相比 FL-Defender, ConTrust 的平均 ASR 进一步降低 0.77 个百分点, 同时平均 ACC 提升了 0.46 个百分点。结果表明, ConTrust 在多种数据集和攻击场景下均具有较稳定的防御表现。

如表 3 所示, FedAvg 在各类攻击下均表现出较高 ASR, 说明传统平均聚合在 Non-IID 与高比例恶意客户端条件下难以有效抑制恶意更新。以 CIFAR-10 数据集下的 FCBA 攻击为例, FedAvg 的 ASR 达到

62.74%，明显高于所有防御方法，与 FedAvg 相比，Multi-Krum、FLTrust、FLDetector 和 FLAIR 均能在一定程度上降低 ASR，但防御稳定性存在差异。Multi-Krum 主要依赖客户端更新之间的全局距离进行筛选，在 Non-IID 场景下容易受到良性异质更新的影响，因此在 DBA、Neurotoxin 和 FCBA 等攻击下仍保持较高 ASR。FLTrust 在部分场景中能够保持较好 ACC，但面对更强的攻击时，其 ASR 仍相对偏高。FLDetector 和 FLAIR 引入历史行为或信誉提升了鲁棒性，在多数场景下优于单轮距离型方法，但仍可能受到局部良性更新差异的影响。

相比之下，HistTrust 在多数攻击场景下同时取得了较低 ASR 和较稳定 ACC。在 MNIST 上，HistTrust 在 DBA、Neurotoxin 和 FCBA 三类攻击下均取得最低 ASR，分别为 22.64%、16.88% 和 26.73%；在标签翻转攻击下，其 ASR 为 10.42%，略高于 FLAIR 的 9.36%，但仍显著低于所有其他方法。在 FMNIST 上，HistTrust 在标签翻转、DBA 和 FCBA 攻击下均取得最低 ASR，并在 Neurotoxin 场景下获得最高 ACC（79.26%），表明其在攻击抑制和主任务性能之间取得了更优平衡。在 CIFAR-10 数据集上，HistTrust 在 4 类攻击下均取得最低 ASR 或接近最低 ASR，并在 DBA 和 Neurotoxin 场景下获得最高 ACC，体现了在强异质环境中的稳定防御能力。

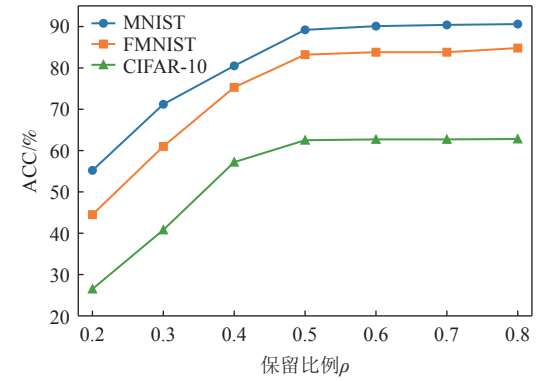
综合表 3 结果可以发现，HistTrust 的主要优势体现在强 Non-IID 和较高恶意客户端比例条件下对攻击的抑制。结果表明，局部模式感知能够降低数据异质性带来的误判风险，历史信誉和自举先验则进一步削弱恶意客户端跨轮持续影响聚合结果的能力。当然，HistTrust 并非在所有场景下都取得最高 ACC，这反映出鲁棒防御与模型精度之间客观存在的权衡关系，但整体防御性能仍优于其他方法。

3）超参数对防御效果的影响

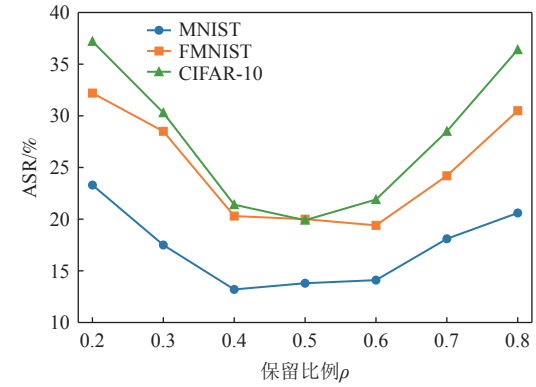
保留比例 ρ 表示根据客户端混合异常分数排序后，保留下来参与最终聚合的低风险客户端比例，是 ConTrust 截断平滑加权聚合策略的关键参数，直接影响 ConTrust 的防御性能与计算开销。为评估保留比例 ρ 对 ConTrust 防御性能的影响，本节将 ρ 分别设置为 0.2、0.3、0.4、0.5、0.7 和 0.8，并在 30% 恶意客户端参与的 FCBA 攻击场景下进行实验。图 3 展示了不同 ρ 取值下 ACC、ASR 和服务器端平均聚合运行时间的变化情况。

结果表明，当 ρ 较小时，由于参与聚合的客户端数量过少，部分正常更新会被误截断，导致模型训练信息不足，从而降低模型准确率。随着ρ增大，更多正常客户端更新被保留，ACC 逐步提升，但恶意更新也更容易混入聚合权重中，导致 ASR 回升。

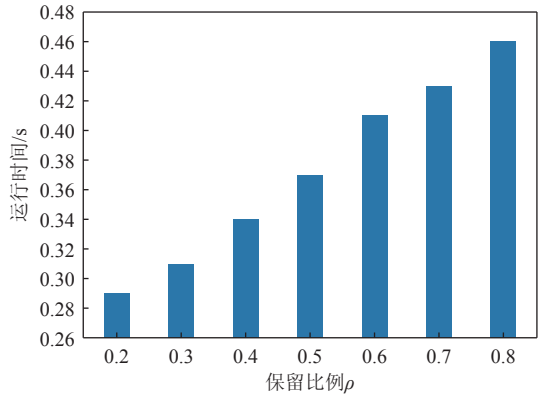
从综合表现看，$\rho = 0.5$ 附近通常能够取得较好的折中效果。该取值既能保留足够的正常客户端用于模型训练，又能有效截断高风险更新，避免恶意客户端通过尾部权重持续影响全局模型。此外，随着 ρ 增大，服务器端需要参与加权聚合的客户端数量增加，每轮平均聚合时间也会略有上升。不过，相较于防御性能变化，运行时间增长整体较为平缓。综合



(a) 保留比例对模型准确率的影响



(b) 保留比例对攻击成功率的影响



(c) 保留比例对服务器平均聚合运行时间的影响

Fig. 3 The influence of the retention ratio ρ on the defense performance of ConTrust

图 3 保留比例 ρ 对 ConTrust 防御性能的影响

考虑攻击成功率、模型准确率和计算开销，本文将 $\rho=0.5$ 作为 ConTrust 的默认设置。

自举先验权重 β 用于调节历史行为先验与当前轮即时信任在自举可信度计算中的融合比例。为评估 β 对 HistTrust 防御性能的影响，本节设计关于 β 的参数敏感性实验，将其取值设置为 β=0.2~0.8，在 30% 恶意客户端参与的 FCBA 攻击场景下，分别在 MNIST、FMNIST 和 CIFAR-10 数据集上开展实验，结果如图 4 所示。

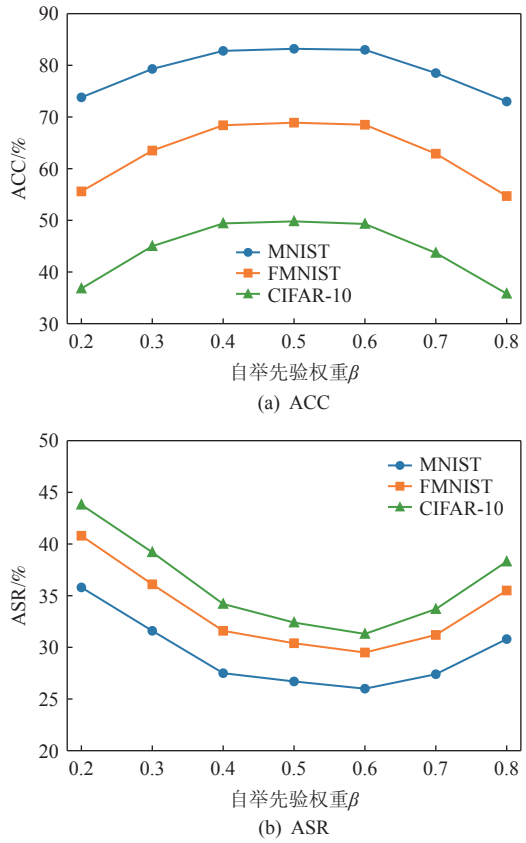


Fig. 4 The influence of the self-estimated prior weight β on the defense performance of HistTrust

图 4 自举先验权重 β 对 HistTrust 防御性能的影响

由图 4 可见，β 对 HistTrust 的防御性能具有较明显影响。当 β 较小时，模型主要依赖当前轮即时信任，历史行为信息利用不足，难以充分发挥跨轮累积判别优势，对攻击的抑制能力较弱；当 β 较大时，聚合过程过度依赖历史信誉，对当前轮正常更新变化的响应能力下降，在数据异质性较强时容易削弱部分良性客户端的训练贡献，从而影响模型精度。相比之下，$\beta=0.5$时历史先验与即时信任能够形成较好的协同作用，在保持较高 ACC 的同时有效降低 ASR，综合防御性能最佳，因此默认将 β 设置为 0.5。

为进一步分析数据异质性对 HistTrust 防御性能的影响，将 Dirichlet 参数 α 分别设置为 0.1、0.3、0.5、0.7 和 0.9，同样在 30% 恶意客户端参与的 FCBA 攻击场景下，于 MNIST、FMNIST 和 CIFAR-10 数据集上开展实验，结果如 5 所示。结果表明，随着 α 逐渐增加，3 个数据集上的 ACC 整体呈上升趋势，ASR 总体呈下降趋势。由于 α 越小，客户端数据分布越不均衡，良性客户端更新之间天然存在更大的方向和幅值差异，恶意更新更容易借助正常异质性进行伪装，从而增加服务器端可信评估的难度；随着 α 增大，客户端数据分布逐渐接近 IID，良性更新的一致性增强，可信评估基线更加稳定，因此模型能够获得更高的 ACC 和更低的 ASR。

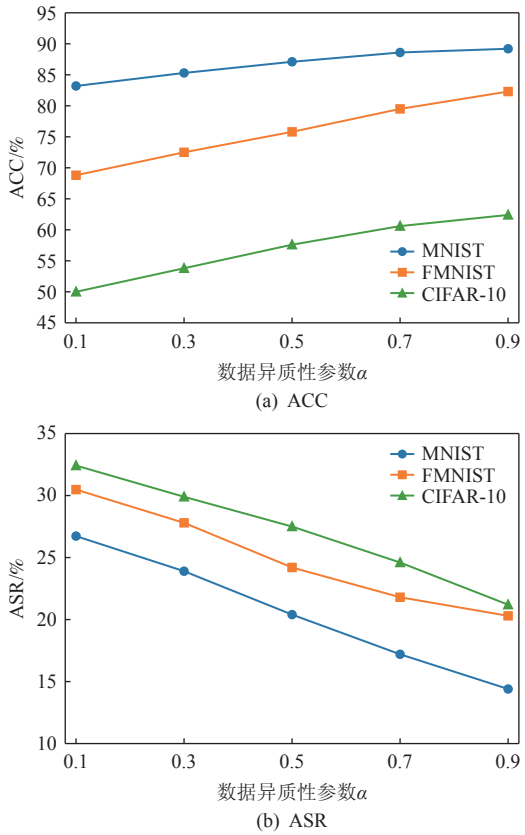


Fig. 5 The influence of the Dirichlet parameter α on the defense performance of HistTrust

图 5 Dirichlet 参数 α 对 HistTrust 防御性能的影响

4）消融实验

为进一步验证 ConTrust 各组成模块对防御性能的贡献，本节在 30% 恶意客户端比例下开展消融实验。完整 ConTrust 同时包含方向一致性评估（Dir）、主子空间谱残差评估（Spec）和截断保留机制（Trunc）。本节构造 ConTrust-w/o-Dir、ConTrust-w/o-Spec 和 ConTrust-w/o-Trunc 三种变体，并在标签翻转、DBA、Neurotoxin 和 FCBA 四类攻击下开展实验，结

果如图 6 所示。

具体来看, ConTrust-w/o-Dir 在多种攻击场景下的 ASR 均明显升高。这表明仅依赖幅值偏离与谱残差, 难以充分识别更新方向存在显著偏移的恶意客户端。而方向一致性项能够有效刻画客户端是否偏离良性更新的优化趋势。去除方向一致性后, 服务器对恶意方向扰动的敏感性下降, 防御能力随之减弱。

对于 ConTrust-w/o-Spec, 其防御效果明显下降。在 CIFAR-10 的 FCBA 攻击下, ConTrust-w/o-Spec 的 ASR 为 26.37%, 而完整 ConTrust 降低到 19.95%。这说明谱残差能够有效检测更新在主子空间之外的异常扰动, 提供额外判别能力。

对于 ConTrust-w/o-Trunc, 其在多数场景下均表现出最高的 ASR, 说明仅依赖连续权重分配仍难以完全抑制高风险客户端对全局模型的干扰。相比之下, 完整 ConTrust 采用“先截断、再平滑加权”的策略, 能够先去除高风险客户端, 再在低风险集合中分配连续权重, 因此具有更强鲁棒性。

综上, 消融实验充分验证了 ConTrust 各组成模块的贡献。其中方向一致性项增强了对恶意方向偏移的识别能力, 谱残差项提升了对高维结构异常的检测能力, 截断保留机制强化了聚合阶段对高风险客户端的隔离效果。上述 3 个核心模块在防御过程中发挥互补作用, 使完整 ConTrust 在不同攻击场景下均能保持较高 ACC 和较低 ASR。

同样地, 为验证 HistTrust 各组成模块对防御性能的贡献, 构造 3 种消融变体进行对比实验。HistTrust-w/o-Local: 去除局部模式感知, 仅保留非局部化的当前轮评分; HistTrust-w/o-Hist: 去除历史信誉累积, 仅依据当前轮信任进行聚合; HistTrust-w/o-Boot: 去除自举式融合机制, 不再将即时信任与历史先验进行联合建模。实验同样在 3 类数据集和 4 类攻击场景下开展, 结果如图 7 所示。

HistTrust-w/o-Local 在多数场景下的 ASR 明显升高, 尤其在 DBA 和 FCBA 攻击中表现更为明显。

例如, 在 CIFAR-10 的 FCBA 攻击下, ASR 达到 64.85%, 而完整 HistTrust 降至 32.42%。这表明, 当客户端数据分布存在显著异质性时, 仅依赖全局统计特征难以准确区分正常更新与恶意更新, 而局部模式感知能够为客户端建立更加合理的可信评估基准, 有效降低数据异质性带来的误判风险。

HistTrust-w/o-Hist 的 ASR 低于 HistTrust -w/o-Local, 但仍明显高于完整 HistTrust。这表明仅依赖当前轮局部可信评估能够识别部分异常更新, 但缺乏跨轮行为建模时, 持续性或间歇性攻击更容易规避单轮检测。历史信誉累积能够记录客户端跨轮行

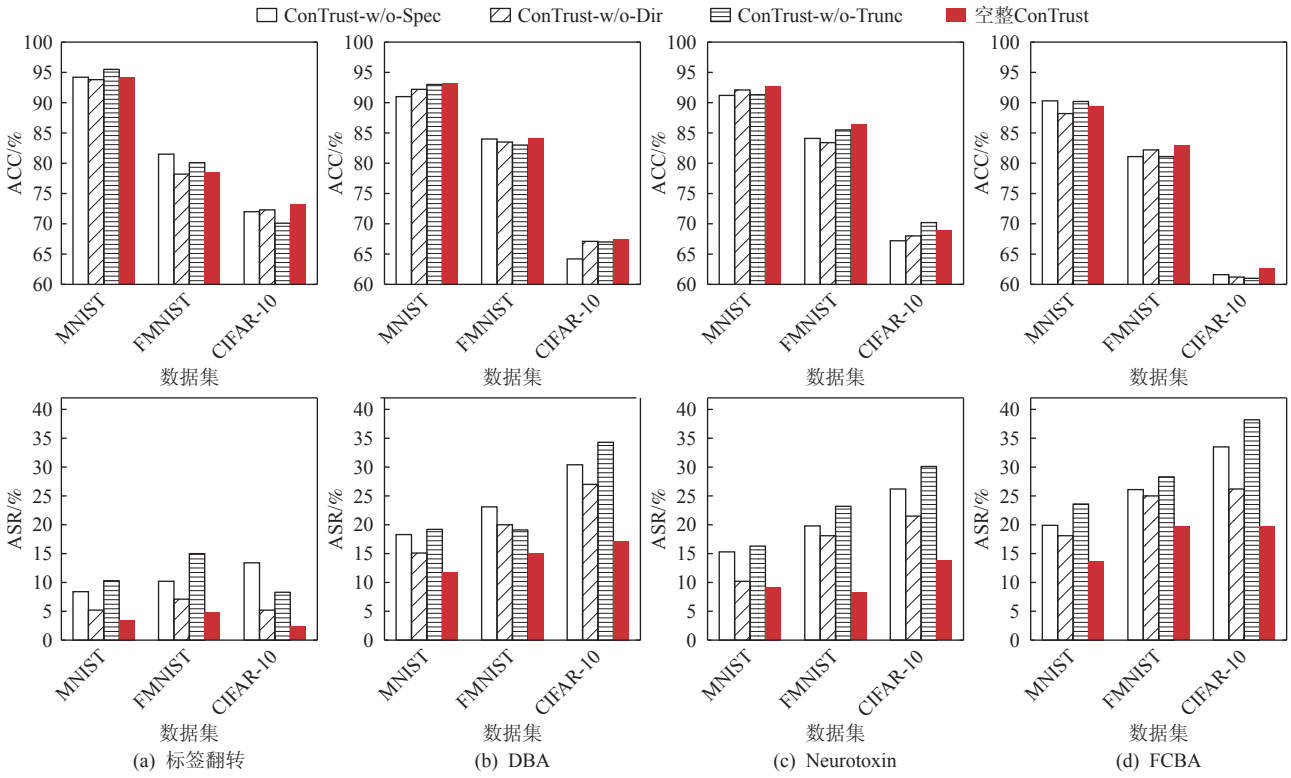


Fig. 6 Ablation Studies of ConTrust

图 6 ConTrust 消融实验

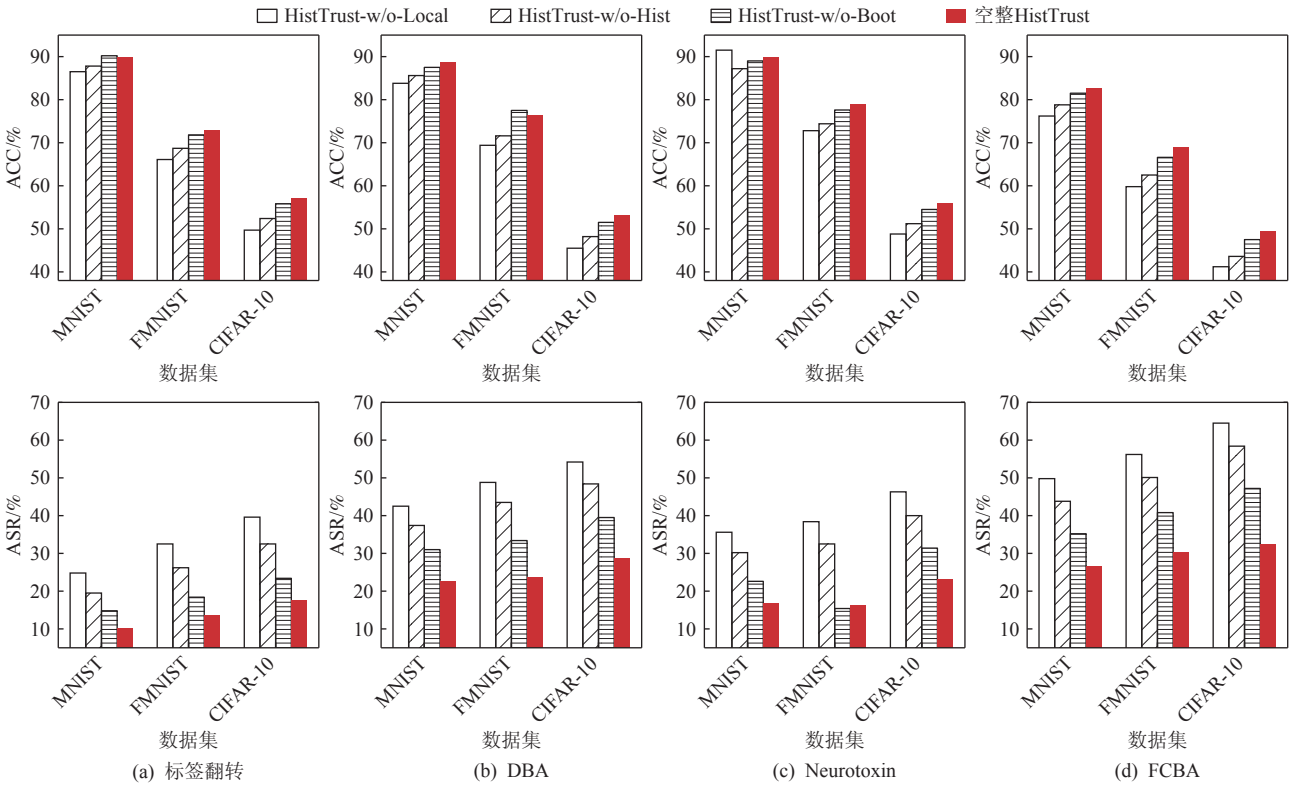


Fig. 7 Ablation Studies of HistTrust

图 7 HistTrust 消融实验

为，有效增强了对长期异常行为的识别能力。

HistTrust-w/o-Boot 在部分场景下能保持较高 ACC，但 ASR 仍显著高于完整方法。例如，在 FMNIST 的 DBA 攻击下，HistTrust-w/o-Boot 的 ACC 为 77.72%，略高于完整方法的 76.84%，但其 ASR 达到 33.58%，明显高于完整方法的 23.72%。这说明自举式融合机制能充分结合历史先验与当前评估，在模型性能和攻击抑制能力之间取得更好平衡。

综上，消融实验验证了 HistTrust 各模块的防御贡献。其中，局部模式感知降低 Non-IID 下的误判风险，历史信誉累积增强对跨轮攻击的识别能力，自举式动态聚合则把当前轮即时信任与历史先验统一为可执行的聚合决策。三者协同作用，使 HistTrust 在不同攻击场景下均能够保持较高 ACC 和较低 ASR。

5 结 论

本文提出了一种面向数据异质性的联邦学习投毒防御框架。针对不同数据分布条件下客户端更新的差异，分别设计了适用于 IID 场景的 ConTrust 和适用于 Non-IID 场景的 HistTrus，构建了统一的防御思路，为数据异质环境下联邦学习投毒防御提供了一种有效解决方案。基于 MNIST、Fashion-MNIST 和 CIFAR-10 数据集，在标签翻转、DBA、Neurotoxin 和 FCBA 等典型投毒攻击场景下开展实验验证。实验结果表明，所提防御框架在不同数据分布条件下均能够有效降低 ASR，同时保持较高的 ACC，综合防御性能优于现有防御方法，验证了所提框架在复杂数据异质环境下的有效性与鲁棒性。

随着动态异构网络中海量异质终端的持续接入，联邦学习面临更加复杂的数据分布和安全威胁。未来将进一步研究更加复杂数据异质条件下的防御方法，以及面向客户端动态加入与退出、多类型协同攻击和大模型联邦学习等场景的投毒防御技术，不断提升联邦学习系统在真实动态环境中的安全性与鲁棒性。

作者贡献声明：周辉、陈昊负责方案设计、实验验证，并撰写论文；秦拯、刘羽提出算法思路，并修改论文。

参 考 文 献

[1] Feng Yiming, Qian Zhen, Li Guanghui, et al. Collaborative

optimization method for adaptive hierarchical federated learning in heterogeneous edge environments[J]. Journal of Computer Research and Development, 2025, 62(6): 1416−1433 (in Chinese)
(冯奕铭, 钱珍, 李光辉, 等. 异构边缘环境下自适应分层联邦学习协同优化方法[J]. 计算机研究与发展, 2025, 62(6): 1416−1433)

[2] Le Junqing, Tan Zhouyong, Zhang Di, et al. Secure and efficient federated learning for continuous IoV data sharing[J]. Journal of Computer Research and Development, 2024, 61(9): 2199−2212 (in Chinese)
(乐俊青, 谭州勇, 张迪, 等. 面向车联网数据持续共享的安全高效联邦学习[J]. 计算机研究与发展, 2024, 61(9): 2199−2212)

[3] McMahan B, Moore E, Ramage D, et al. Communication-efficient learning of deep networks from decentralized data[C]//Proc of the 20th Int Conf on Artificial Intelligence and Statistics. Fort Lauderdale, USA: PMLR, 2017: 1273−1282

[4] Zhou Hui, Qin Zheng, Sun Peng, et al. Unifying gradient leakage attacks against privacy-protected federated learning in IoT networks[J]. IEEE Internet of Things Journal, 2026, 13(4): 6564−6574

[5] Liu Jialang, Guo Yanming, Lao Mingrui, et al. A survey of backdoor attack and defense algorithms in federated learning[J]. Journal of Computer Research and Development, 2024, 61(10): 2607−2626 (in Chinese)
(刘嘉浪, 郭延明, 老明瑞, 等. 基于联邦学习的后门攻击与防御算法综述[J]. 计算机研究与发展, 2024, 61(10): 2607−2626)

[6] Zhang Xiaojian, Zhang Leilei, Zhang Zhizheng. Federated learning method based on user-level local differential privacy[J]. Journal of Computer Research and Development, 2025, 62(2): 472−487 (in Chinese)
(张啸剑, 张雷雷, 张治政. 基于用户级别本地化差分隐私的联邦学习方法[J]. 计算机研究与发展, 2025, 62(2): 472−487)

[7] Hong Zhen, Feng Wanglei, Wen Zhenyu, et al. Detecting free-riding attack in federated learning based on gradient backtracking[J]. Journal of Computer Research and Development, 2024, 61(9): 2185−2198 (in Chinese)
(洪榛, 冯王磊, 温震宇, 等. 基于梯度回溯的联邦学习搭便车攻击检测[J]. 计算机研究与发展, 2024, 61(9): 2185−2198)

[8] Ma Mengyao, Liu Shuofeng, Vo V, et al. ClieND: Client-side neuron-level detection against poisoning attacks on cross-silo federated learning[C]//Proc of the ACM Asia Conf on Computer and Communications Security. New York: ACM, 2026: 1108−1124

[9] Sharma A, Chen W, Zhao J C, et al. FLAIR: Defense against model poisoning attack in federated learning[C]//Proc of the 18th ACM Asia Conf on Computer and Communications Security. New York: ACM, 2023: 553−566

[10] Cao Xiaoyu, Jia Jinyuan, Zhang Zaixi, et al. FedRecover: Recovering from poisoning attacks in federated learning using historical information[C]//Proc of the 2023 IEEE Symp on Security and Privacy. Piscataway, NJ: IEEE, 2023: 1366−1383

[11] Blanchard P, El Mhamdi E M, Guerraoui R, et al. Machine learning with adversaries: Byzantine tolerant gradient descent[C]//Advances in Neural Information Processing Systems. Long Beach, USA: NeurIPS, 2017: 118−128

[12] Yin Dong, Chen Yudong, Kannan R, et al. Byzantine-robust distributed learning: Towards optimal statistical rates[C]//Proc of the 35th Int Conf on Machine Learning. Stockholm, Sweden: PMLR, 2018: 5650−5659

[13] El Mhamdi E M, Guerraoui R, Rouault S. The hidden vulnerability of distributed learning in Byzantium[C]//Proc of the 35th Int Conf on Machine Learning. Stockholm, Sweden: PMLR, 2018: 3521−3530

[14] Pillutla K, Kakade S M, Harchaoui Z. Robust aggregation for federated learning[J]. IEEE Transactions on Signal Processing, 2022, 70: 1142−1154

[15] Fung C, Yoon C J M, Beschastnikh I. The limitations of federated learning in sybil settings[C]//Proc of the 23rd Int Symp on Research in Attacks, Intrusions and Defenses. Berkeley, USA: USENIX Association, 2020: 301−316

[16] Ozdayi M S, Kantarcioglu M, Gel Y R. Defending against backdoors in federated learning with robust learning rate[C]//Proc of the AAAI Conf on Artificial Intelligence. Palo Alto, USA: AAAI, 2021, 35(10): 9268−9276

[17] Cao Xiaoyu, Fang Minghong, Liu Jia, et al. FLTrust: Byzantine-robust federated learning via trust bootstrapping[C]//Proc of the Network and Distributed System Security Symp. Reston, USA: Internet Society, 2021: 1−18

[18] Nguyen T D, Rieger P, Chen H, et al. FLAME: Taming backdoors in federated learning[C]//Proc of the 31st USENIX Security Symp. Berkeley, USA: USENIX Association, 2022: 1415−1432

[19] Zhang Zaixi, Cao Xiaoyu, Jia Jinyuan, et al. FLDetector: Defending federated learning against model poisoning attacks via detecting malicious clients[C]//Proc of the 28th ACM SIGKDD Conf on Knowledge Discovery and Data Mining. New York: ACM, 2022: 2545−2555

[20] Ma Xindi, Li Qinghua, Jiang Qi, et al. Byzantine robust federated learning for Non-IID data[J]. Journal on Communications, 2023, 44(6): 138−153 (in Chinese)
(马鑫迪, 李清华, 姜奇, 等. 面向 Non-IID 数据的拜占庭鲁棒联邦学习[J]. 通信学报, 2023, 44(6): 138−153)

[21] Biggio B, Nelson B, Laskov P. Poisoning attacks against support vector machines[C]//Proc of the 29th Int Conf on Machine Learning. Edinburgh, UK: Omnipress, 2012: 1807−1814

[22] Muñoz-González L, Biggio B, Demontis A, et al. Towards poisoning of deep learning algorithms with back-gradient optimization[C]//Proc of the 10th ACM Workshop on Artificial Intelligence and Security. New York: ACM, 2017: 27−38

[23] Tolpegin V, Truex S, Gursoy M E, et al. Data poisoning attacks against federated learning systems[C]//Proc of the 25th European Symp on Research in Computer Security. Cham: Springer, 2020: 480−501

[24] Baruch M, Baruch G, Goldberg Y. A little is enough: Circumventing defenses for distributed learning[C]//Advances in Neural Information Processing Systems. Vancouver, Canada: NeurIPS, 2019: 8635−8645

[25] Fang Minghong, Cao Xiaoyu, Jia Jinyuan, et al. Local model poisoning attacks to Byzantine-robust federated learning[C]//Proc of the 29th USENIX Security Symp. Berkeley, USA: USENIX Association, 2020: 1605−1622

[26] Xie Cong, Koyejo O, Gupta I. Fall of empires: Breaking Byzantine-tolerant SGD by inner product manipulation[C]//Proc of the 35th Uncertainty in Artificial Intelligence Conf. Tel Aviv, Israel: PMLR, 2020: 261−270

[27] Shejwalkar V, Houmansadr A. Manipulating the Byzantine: Optimizing model poisoning attacks and defenses for federated learning[C]//Proc of the Network and Distributed System Security Symp. Reston, USA: Internet Society, 2021: 1−18

[28] Zhang Guifen, Su Qinchun, Li Hongzi, et al. Federated learning adaptive gradient accumulation backdoor attack[J]. Journal of Computer Research and Development, 2026: 1−15[2026-07-06]. https://link.cnki.net/urlid/11.1777. tp.20260515.1455.008 (in Chinese)
(张桂芬, 苏秦春, 李鸿梓, 等. 联邦学习自适应梯度累积后门攻击[J/OL]. 计算机研究与发展, 2026: 1−15[2026-07-06]. https://link.cnki.net/urlid/11.1777. tp.20260515.1455.008)

[29] Lü Xiaoting, Liu Jingkai, Liu Zhichen, et al. A survey of backdoor attack threats and adversarial defense methods in federated learning[J]. Chinese Journal of Computers, 2025, 48(12): 2823−2854 (in Chinese)
(吕晓婷, 刘敬楷, 刘芷辰, 等. 联邦学习后门攻击威胁与对抗性防御方法综述[J]. 计算机学报, 2025, 48(12): 2823−2854)

[30] Bagdasaryan E, Veit A, Hua Yiqing, et al. How to backdoor federated learning[C]//Proc of the 23rd Int Conf on Artificial Intelligence and Statistics. Palermo, Italy: PMLR, 2020: 2938−2948

[31] Xie Chunlin, Huang Keli, Chen Pinyu, et al. DBA: Distributed backdoor attacks against federated learning[C]//Proc of Int Conf on Learning Representations. Addis Ababa, Ethiopia: ICLR, 2020: 1−19

[32] Wang Hongyi, Sreenivasan K, Rajput S, et al. Attack of the tails: Yes, you really can backdoor federated learning[C]//Advances in Neural Information Processing Systems. Vancouver, Canada: NeurIPS, 2020: 16070−16084

[33] Zhang Zhengming, Panda A, Song Linyue, et al. Neurotoxin: Durable backdoors in federated learning[C]//Proc of the 39th Int Conf on Machine Learning. Baltimore, USA: PMLR, 2022: 26429−26446

[34] Dai Yanbo, Li Songze. Chameleon: Adapting to peer images for planting durable backdoors in federated learning[C]//Proc of the 40th Int Conf on Machine Learning. Honolulu, USA: PMLR, 2023: 6712−6725

[35] Lyu Xiaoting, Han Yufei, Wang Wei, et al. Poisoning with Cerberus: Stealthy and colluded backdoor attack against federated learning[C]//Proc of the AAAI Conf on Artificial Intelligence. Washington, USA: AAAI, 2023, 37(7): 9020−9028

[36] Zhuang Haomin, Yu Mingxian, Wang Hao, et al. Backdoor federated learning by poisoning backdoor-critical layers[C]//Proc of Int Conf on Learning Representations. Vienna, Austria: ICLR, 2024: 40241−40265

[37] Lyu Xiaoting, Han Yufei, Wang Wei, et al. Lurking in the shadows: Unveiling stealthy backdoor attacks against personalized federated learning[C]//Proc of the 33rd USENIX Security Symp. Berkeley, USA: USENIX Association, 2024: 4157−4174

[38] Liu Tao, Zhang Yuhang, Feng Zhu, et al. Beyond traditional threats: A persistent backdoor attack on federated learning[C]//Proc of the AAAI Conf on Artificial Intelligence. Palo Alto, USA: AAAI, 2024, 38(19): 21359−21367

[39] Han S, Park S, Wu Fangzhou, et al. Towards attack-tolerant federated learning via critical parameter analysis[C]//Proc of the IEEE/CVF Int Conf on Computer Vision. Paris, France: IEEE, 2023: 4985−4994

[40] LeCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278−2324

[41] Xiao Han, Rasul K, Vollgraf R. Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms[EB/OL]. 2017[2026-07-06]. https://arxiv.org/abs/1708.07747.

[42] Krizhevsky A. Learning multiple layers of features from tiny images[R]. Toronto: University of Toronto, 2009

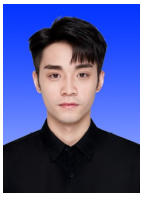
[43] Jebreel N, Domingo-Ferrer J. FL-Defender: Combating targeted attacks in federated learning[J]. Knowledge-Based Systems, 2023, 260: 110178

[44] Xu Hairuo, Shu Tao. Defending against model poisoning attack in federated learning: A variance-minimization approach[J]. Journal of Information Security and Applications, 2024, 82: 103744



Zhou Hui, born in 1999. PhD candidate. Member of CCF. His main research interest includes federated learning.

周　辉, 1999 年生。博士研究生。CCF 学生会员。主要研究方向为联邦学习。



Chen Hao, born in 2000. Master candidate. His main research interest includes federated learning.

陈　昊, 2000 年生。硕士研究生。主要研究方向为联邦学习。



Qin Zheng, born in 1969. PhD, professor. Member of CCF. His main research interests include data security and privacy protection, and LLMs security.

秦　拯, 1969 年生。博士, 教授。CCF 会员。主要研究方向为数据安全与隐私保护、大模型安全。



Liu Yu, born in 1994. PhD, assistant professor. Member of CCF. Her main research interests include IoT security, data security, and artificial intelligence security.

刘　羽, 1994 年生。博士, 助理教授。CCF 会员。主要研究方向为物联网安全、数据安全、人工智能安全。