基于可解释贡献异常检测与动态裁剪的联邦学习投毒攻击防御方法

蒋伟进[1),2)] 杨 璇[1),2)] 李碧霞[1),2)]

[1)](湖南工商大学计算机学院 长沙 410205)

[2)](湘江实验室 长沙 410205)

摘 要 联邦学习允许各参与方在不暴露本地数据的情况下协同训练模型,但在实际应用中仍面临投毒攻击等安全威胁。现有主流方法通常采用异常检测和鲁棒聚合相结合的方法应对这类威胁。然而,这类方法易将迭代训练中良性客户端参数更新的正常异质性识别为异常,并且在剔除恶意客户端时,往往会舍弃其更新中潜在的良性信息,导致全局模型性能损失。为解决这一问题,本文提出了一种基于可解释贡献异常检测与动态裁剪(Contribution Anomaly Detection and Clipping,CADC)算法。该算法通过结合SHAP值和局部异常因子指标(Local Contribution Outlier Factor,LCOF),量化客户端参数更新对模型预测行为的贡献,识别出偏离正常行为的恶意客户端,并对识别出的恶意更新进行动态裁剪,保留其中对全局模型有积极贡献的参数。实验结果表明,即使在数据为非独立同分布或恶意客户端比例较高的情况下,CADC方法依然能够以高达90%的准确率有效识别恶意客户端。与主流联邦学习投毒攻击防御方法相比,CADC在MNIST数据集上的恶意客户端识别准确率提升2%~23.5%,同时全局模型的预测准确率提升1.98%~17.41%。

关键词 联邦学习;投毒攻击;异常检测;动态裁剪;模型鲁棒性

中图法分类号 TP18 **DOI号** 10.11897/SP.J.1016.2025.02855

Federated Learning Poisoning Attack Defense Method Based on Interpretable Contribution Anomaly Detection and Dynamic Pruning

JIANG Wei-Jin[1),2)] YANG Xuan[1),2)] LI Bi-Xia[1),2)]

[1)](School of Computer Science, Hunan University of Technology and Business, Changsha 410205)

[2)](Xiangjiang Laboratory, Changsha 410205)

Abstract Federated Learning (FL) is a decentralized machine learning paradigm that enables multiple clients to collaboratively train a shared global model without exposing their local data. This privacy protection method has been widely applied in sensitive fields such as finance, healthcare, and the Internet of Things. By promoting the use of decentralized data sources, FL has reduced data silos and lowered privacy risks associated with centralized data collection. However, FL's distributed architecture and open training environment make it vulnerable to various security threats, among which poisoning attacks are particularly severe. In such attacks, malicious clients submit carefully designed malicious updates to reduce the performance of the global model or manipulate its predictions, posing significant challenges to the robustness and reliability of FL systems in actual deployment. To prevent model poisoning, mainstream methods

收稿日期:2024-12-27;在线发布日期:2025-07-24。本课题得到国家自然科学基金(No. 61772196)、湖南省教育厅科学研究重点项目(No. 24A0446,No. 24A0753)、湖南工商大学研究生科研创新项目(No. CX2024YB001)资助。**蒋伟进**,博士,教授,博士生导师,中国计算机学会(CCF)高级会员,主要研究领域为边缘计算、社会计算、网络空间安全。E-mail:jwjnudt@163.com。**杨 璇**,硕士,中国计算机学会(CCF)学生会员,主要研究领域为联邦学习、隐私保护。**李碧霞**,硕士,中国计算机学会(CCF)学生会员,主要研究领域为联邦学习、模型安全。

typically combine anomaly detection with robust aggregation techniques to identify and clip malicious updates before aggregation. However, these defenses face two major challenges in real world. Firstly, due to the high heterogeneity of client data (Non-IID), even benign clients may naturally generate different model updates. This increases the risk of false positives, where normal updates are mistakenly marked as malicious, thereby compromising the fairness and generality of the global model. Secondly, most existing defense measures adopt a unified strategy, such as completely discarding or uniformly cutting detected malicious updates, without considering that these updates may still contain some useful information. This indiscriminate processing often leads to the loss of unnecessary useful parameter information, reduced learning efficiency, and decreased model performance. To address these limitations, this paper proposes a new defense framework called Contribution Anomaly Detection and Clipping (CADC). CADC aims to improve the accuracy of attack detection and global model accuracy. It first uses SHAP (Shapley Additive exPlans) values to quantitatively evaluate the contribution of each customer′s model updates to the global model′s predictive behavior. By capturing the marginal impact of each update on the model output, SHAP provides a finegrained and interpretable evaluation of the update quality. Based on these contribution scores, CADC introduces the Local Contribution Outlier Factor (LCOF) algorithm to detect potential malicious updates by identifying clients whose contribution patterns deviate significantly from the known benign update distribution. Unlike traditional defenses that discard all suspicious updates, CADC further applies dynamic pruning algorithms. This algorithm selectively prunes parameters in suspicious malicious updates, while considering the degree of malice of the parameters and their deviation from the benign update reference set. By retaining parameters with positive effects while pruning harmful components, CADC minimizes unnecessary information loss, improves training efficiency, and enhances model robustness to the greatest extent possible. Extensive experiments conducted on Non-IID data settings and different proportions of malicious clients have shown that CADC achieves an accuracy rate of up to 90% in adversarial client detection. On the MNIST dataset, CADC improved detection accuracy by 2% to 23.5% and global model accuracy by 1.98% to 17.41% compared to mainstream defense methods. These results indicate that CADC provides a more adaptive, accurate, and effective solution for defending against model poisoning in federated learning.

Keywords federated learning; poisoning attack; anomaly detection; dynamic cropping; model robustness

1 引 言

联邦学习(Federated Learning,FL)通过允许各参与方在本地训练模型,解决了数据隐私保护与协同建模的冲突[1]。其去中心化特性使其在医疗、环境监测等数据敏感领域具有广泛应用前景[2-5]。然而,由于各客户端本地训练过程的不可见性,恶意参与方可通过篡改本地训练数据或模型更新,干扰全局模型训练,威胁模型安全[6,7]。

研究表明,投毒攻击(Poisoning Attack,PA)已成为联邦学习系统面临的主要安全威胁之一[8-9]。恶意客户端通过注入恶意数据或操控本地模型更新,导致全局模型性能下降,甚至在特定任务中输出错误预测结果。当前主流防御策略多采用“检测-裁剪”的两阶段方法:首先基于模型更新之间的相似度识别异常客户端[10,11],随后对其更新参数进行均匀裁剪,以减少其对全局模型的干扰[12]。

然而,该类方法存在两个关键问题:其一,固定阈值检测方法难以适应联邦学习的动态训练过程。在非独立同分布(Non Independent and Identically Distributed,Non-IID)数据下,迭代训练会引发良性更新的自然发散[13];同时,高比例恶意客户端会破坏良性更新的统计主导性,导致其被误判为异常[14,15]。

这类方法依赖整体更新向量的相似度，难以区分数据异质性引起的正常偏移与恶意扰动导致的异常更新，进而引发误报。其二，均匀裁剪策略可能损害恶意更新中包含的有益信息。攻击者的更新通常由恶意扰动与正常成分混合构成，若将异常更新整体裁剪，会丢弃其中良性成分，造成收敛效率与全局模型性能的损失[16]。

为应对上述问题，本文提出一种基于可解释性分析的防御方法。该方法将可解释性技术提供的参数级贡献信息，作为细粒度异常检测依据，通过分析各参数对模型输出的局部影响，识别出在参数维度上表现异常的更新[17]，从而有效区分由Non-IID数据引起的正常发散与由投毒攻击导致的异常扰动，避免在高比例恶意客户端环境下将良性更新误判为异常。此外，该方法进一步利用可解释性技术对恶意更新每个参数的“恶意程度”进行量化评估，实现参数级别的动态裁剪，最大程度保留其中的良性成分。本研究的主要贡献包括：

(1)提出联邦学习中基于参数级可解释性的异常检测算法，通过构建参数贡献矩阵与LCOF指标，实现细粒度恶意客户端识别；

(2)设计参数级动态裁剪算法，结合LCOF量化异常程度并选择性保留潜在良性参数，突破传统整体裁剪的局限性；

(3)通过多组实验验证了所提方法在高比例恶意客户端和Non-IID环境下的有效性。实验结果表明，该方法能够以超过90%的准确率识别恶意客户端，并提升全局模型准确率，为联邦学习系统中的投毒攻击防御提供了实践参考。

2　相关工作

2.1　联邦学习投毒攻击

在联邦学习系统中，投毒攻击通常分为两类：数据投毒攻击(Data Poisoning Attack，DPA)与模型投毒攻击(Model Poisoning Attack，MPA)。

DPA通过篡改本地数据集，在训练阶段注入误导性信息，干扰模型训练[18]。Gupta等[19]提出一种基于良性梯度正交方向的攻击方法，在梯度空间中生成与正常梯度相互独立的干扰项，实现在数据和模型更新层面的投毒攻击。Kasyap等[20]将原始输入映射至高维语义空间，并在目标类别附近施加扰动，实现对模型决策边界的操控。Tolpegin等[21]提出的标签翻转攻击通过更改部分样本标签，诱导模型学习错误的判别边界。该方法操作简单，且在攻击比例较小的条件下也可显著影响模型预测能力，因而成为常见的投毒方法之一。

MPA是一种通过篡改本地模型更新，影响全局模型参数分布的攻击方式[22-24]。Fang等[25]提出的局部模型投毒攻击，通过持续操控受控客户端，在多个通信轮次中注入恶意更新，在不引发显著异常波动的情况下逐步削弱全局模型的收敛性，从而规避基于瞬时偏差的检测机制。Bagdasaryan等[26]提出的缩放攻击将检测约束整合至攻击者损失函数，通过缩小恶意更新向量的范数以规避鲁棒聚合机制的检测。Cao等[27]探讨了虚假客户端注入策略，攻击者通过伪造多个客户端身份协同上传恶意更新，在聚合阶段放大攻击影响，干扰全局模型训练。Zhang等[28]提出了一种基于生成对抗网络的攻击框架，攻击者利用全局模型的迭代参数重构其他参与者的数据样本特征，从而实施更具针对性的模型劫持和数据泄露。

2.2　投毒攻击防御方法

面对联邦学习系统遭受的投毒攻击威胁，研究者从相似度与鲁棒聚合、客户端可信度建模以及客户端行为结构建模等多个维度提出了一系列防御策略。

在相似度驱动的鲁棒聚合策略方面，Blanchard等[29]提出的Krum算法，通过计算客户端更新之间的欧氏距离，选择最接近多数更新的客户端参与聚合，以剔除异常值。Xie等[30]提出的Zeno方法引入方向得分，以衡量各更新对全局损失下降的影响，并据此调整聚合权重，提升模型鲁棒性。Liu等[31]将共线性掩码与余弦相似度相结合，实现在不泄露私有数据的前提下的异常检测。Lai等[32]结合验证集性能与更新相似度，增强了攻击识别的准确性。这类方法在低攻击强度或独立同分布数据场景下表现较好，但本质上依赖静态的全局相似度度量，难以适应联邦学习训练过程中的动态变化。尤其在Non-IID场景下，良性客户端更新间的自然发散易被误判为异常。

针对Non-IID场景下的误判问题，Andrew等提出了Clipping方法[33]，通过裁剪更新的模长缓解极端值干扰，并结合差分隐私机制评估参数贡献，提升对异常更新的适应能力。该方法在一定程度上降低了因模长差异导致的误剪风险，但其忽略了更新方向及其对应特征的语义信息，难以准确定位微扰式攻击的关键维度，导致在复杂攻击场景下的防御效

果受限。

在客户端可信度建模方面，Cao等[34]提出FLTrust，通过构建小规模可信数据集训练根模型，进而为各客户端赋予信任评分。郭晶晶等[35]利用模型水印破坏程度判断更新的可信性。Purohit等[36]提出的DataDefense则依赖外部验证数据训练毒性检测器。这类方法虽然在特定设定下有效，但普遍依赖可信数据或先验知识，在实际联邦学习环境中，由于数据不可得性与任务异质性，难以广泛适用。

在客户端行为的结构建模与异常检测方面，Yan等[37]提出FedRoLa，通过模型层级的动态相似性分析，衡量客户端更新与全局结构之间的一致性，从而识别具有异常行为的参与方。Huang等[38]在FDCR中引入Fisher信息矩阵，用于刻画各参数维度的重要性，并基于关键维度的偏移模式进行聚类识别。这类方法通常将客户端整体或模型层作为分析单元，通过比较行为相似性或结构一致性检测异常。然而，在实际攻击中，许多隐蔽策略仅在少数关键参数上引入微小扰动，整体模型结构变化不明显，难以在宏观层面被察觉。基于客户端或层级特征的聚合策略因而无法捕捉此类局部参数级投毒行为，难以在保留潜在良性信息的同时有效剔除恶意扰动。

现有防御方法普遍依赖静态相似度度量，并以客户端为分析单元，导致难以有效检测局部参数层面的扰动攻击。此外，针对识别出的恶意模型，现有方案常采用均匀裁剪策略，在消除恶意参数时，也破坏了恶意模型中的良性参数。

2.3 可解释性技术

可解释性技术的目标在于揭示机器学习模型的内部决策机制与过程，从而提升模型透明度与可信度[39,40]。现有的可解释性方法包括特征重要性分析、局部解释（如LIME、Shapley Value）以及全局解释（如基于决策树、规则提取）[41]。在联邦学习环境中，可解释性技术主要发挥三大作用：(1)量化各客户端对聚合模型的贡献[42]；(2)分析参数贡献与模型行为，检测异常模型更新以识别恶意客户端[43]；(3)在攻击发生后，进行攻击溯源并制定针对性防御方法[44]。

本文采用的SHAP值与Shapley值源于相同的数学理论，但适用范围有所不同。在联邦学习场景下，Shapley值主要用于评估各客户端对全局模型的整体贡献，而SHAP值则针对参数级别进行分解，刻画不同客户端参数对模型决策的影响。例如针对全局模型贡献评估，Song等[45]提出基于Shapley值的贡献指数，以衡量各客户端对联邦训练的贡献，并利用训练过程中的中间结果近似重构模型，从而降低计算成本。Liu等[46]在此基础上提出指导截断梯度的Shapley方法。该方法通过梯度更新重建联邦学习模型，并结合轮内与轮间截断的引导蒙特卡罗抽样，降低了模型重建与贡献评估的计算开销。针对模型异常检测，SHAP被用于提升联邦学习系统的安全性。Sandeepa等[47]利用SHAP计算各客户端上传模型参数的贡献，并基于参数归因结果进行聚类分析，以区分恶意客户端与良性客户端。Khuu等人[48]则将SHAP值作为特征输入支持向量机分类器，用以区分恶意与良性客户端。然而，在非独立同分布数据环境下，良性客户端因其独特数据分布产生的SHAP模式存在显著差异，导致基于全局数据训练的分类器难以建立有效的判别边界，从而容易将具有良性但分布独特的客户端误判为恶意客户端。

3 基于可解释贡献异常检测的投毒攻击防御方法

3.1 设计思想

本文的联邦学习系统包含一个中央服务器（Central Server，CS）和n个客户端，记为$C=\{c_1,c_2,\cdots,c_n\}$。CS负责异常检测与模型聚合，并维护一个小型基准数据集D_s用于SHAP值计算。每轮训练中，CS初始化全局模型并分发给所有客户端。客户端$c_i\left(i\in[1,n]\right)$负责在本地利用本地数据集$d_i$训练模型，并将训练后的模型参数上传至CS。CS基于SHAP值构建参数贡献矩阵，通过LCOF算法量化参数异常得分以识别恶意客户端。对于识别出的恶意客户端，CS应用动态裁剪算法处理其模型更新。最终，CS聚合处理后的恶意客户端参数与良性客户端参数，更新全局模型并开启下一轮迭代。系统架构如图1所示，黄色虚线框内为本文主要工作内容。

3.2 攻击模型与设计目标

CADC方法主要应对以下两种攻击：(1)攻击者篡改客户端本地训练数据，注入误导性样本，干扰模型学习过程；(2)攻击者直接操控模型的参数或更新规则，向全局模型引入偏向性行为。由上述两种攻击产生的恶意更新一旦被服务器聚合，都将影

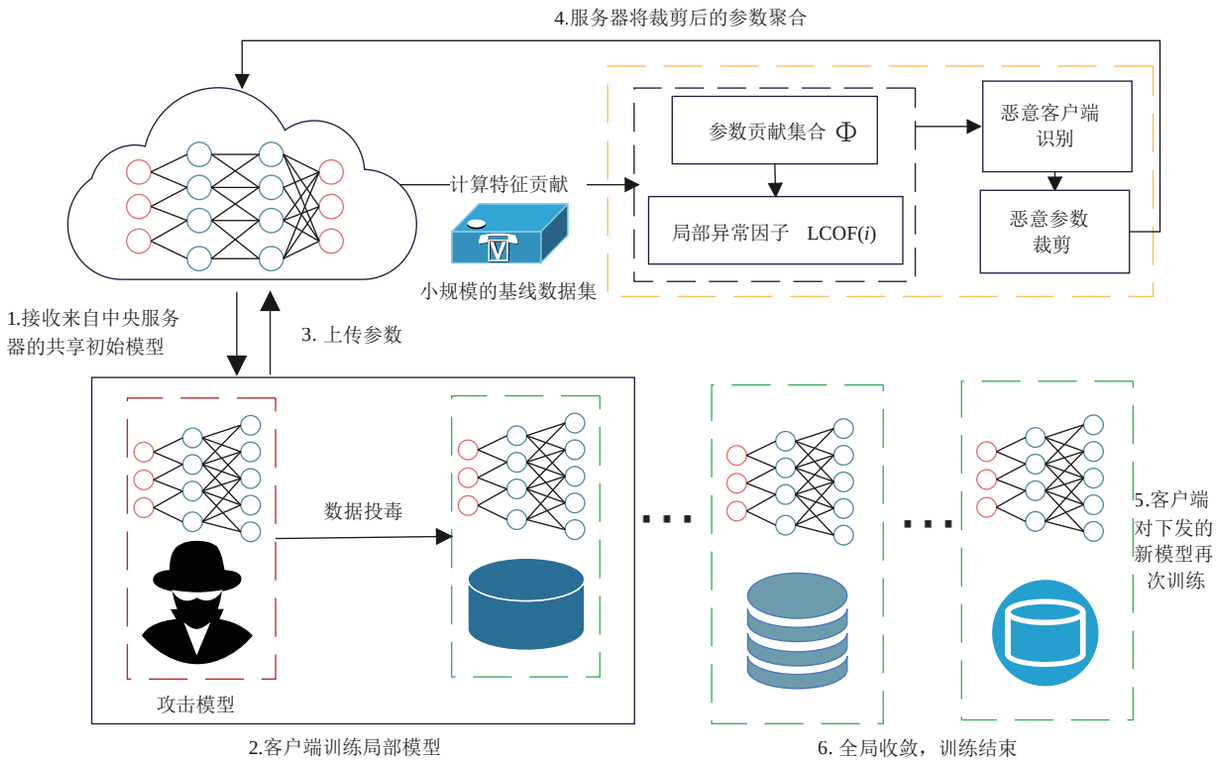


图1　系统架构图

响全局模型的学习方向。为应对上述威胁，本文的设计目标如下：

(1)增强恶意客户端检测能力：基于SHAP值的参数贡献度量与LCOF算法，显著提升Non-IID及高恶意比例场景下的检测准确率；

(2)优化恶意参数处理：通过参数级动态裁剪，抑制攻击影响并保留潜在良性参数，维护全局模型性能。

3.3　详细步骤

本文提出了一种融合恶意客户端识别与参数裁剪的投毒攻击防御方法，主要包括恶意客户端识别算法与动态裁剪算法两大模块。以下将从这两个方面分别展开论述。

(1)恶意客户端识别

由于联邦学习中各客户端的数据分布存在显著异质性，在Non-IID条件下构建统一的SHAP值评估基准面临公平性挑战。为此，本文采用全局参数贡献矩阵及其贡献差异进行度量[47]，从多个类别中聚合SHAP值，缓解因局部数据偏差导致的特定类别主导问题。

对于每个客户端c_i和每个类别$y\in Y$，计算每个参数l对模型输出的SHAP值：

$$\phi_i^{(y)}(l)=\sum_{S\subseteq F\setminus\{l\}}\frac{|S|!\left(|F|-|S|-1\right)!}{|F|!}\left[f\left(x_{S\cup\{l\}}\right)-f\left(x_S\right)\right] \tag{1}$$

其中，$\phi_i^{(y)}(l)$表示参数l对客户端c_i在类别y预测上的贡献值，F为全局模型参数集合，S是参数集合F的一个子集，且不包含参数l，$f\left(x_S\cup\{l\}\right)$表示在参数子集$S$的基础上加入参数$l$后模型的输出，$f\left(x_S\right)$表示仅使用参数集合$S$时模型的输出。

对于每个客户端c_i，将其在所有类别下的参数贡献值进行聚合形成参数贡献集合Φ_i：

$$\Phi_i=\bigcup_{y=1}^{|Y|}\phi_i^{(y)} \tag{2}$$

将所有客户端的参数贡献集合汇总，形成参数贡献矩阵$\boldsymbol{\Gamma}_{tx}$：

$$\boldsymbol{\Gamma}_{tx}=\begin{bmatrix}\Phi_1^{(1)} & \Phi_1^{(2)} & \cdots & \Phi_1^{(c)}\\ \Phi_2^{(1)} & \Phi_2^{(2)} & \cdots & \Phi_2^{(c)}\\ \vdots & \vdots & \ddots & \vdots\\ \Phi_n^{(1)} & \Phi_n^{(2)} & \cdots & \Phi_n^{(c)}\end{bmatrix} \tag{3}$$

用贡献差异评估任意两客户端之间在参数贡献上的差异。对于c_i和c_j，其贡献差异D_{ij}定义为

$$D_{ij}=\left\|\Phi_i^{(y)}-\Phi_j^{(y)}\right\|^2 \tag{4}$$

为识别恶意客户端，本文引入局部贡献密度

$\rho(i)$和局部异常因子LCOF(i)。局部贡献密度$\rho(i)$定义为c_i在其邻域内的贡献密度：

$$\rho(i)=\frac{k}{\sum_{j\in C_k(i)}\text{reachability}-\text{distance}(i,j)} \tag{5}$$

其中，$C_k(i)$是c_i的k个最近邻集合，reachability－distance(i,j)定义为：

$$\text{reachability}-\text{distance}(i,j)=\max\left(D_{ij},k\_\text{distance}(j)\right) \tag{6}$$

$k_\text{distance}(j)$是c_j的第k个最近邻距离。局部异常因子LCOF(i)用于量化c_i相对于其邻居的异常程度，其计算公式为：

$$\text{LCOF}(i)=\frac{1}{\left|C_k(i)\right|}\sum_{j\in C_k(i)}\frac{\rho(j)}{\rho(i)} \tag{7}$$

若LCOF$(i)>1$，表示c_i的贡献密度低于其邻居；若LCOF$(i)\approx1$，表示c_i与其邻居在贡献密度上相似。

基于上述计算，恶意客户端检测步骤如下：

设定阈值θ：阈值θ决定了LCOF值与良性客户端的最大偏差。若某客户端的LCOF值超过该阈值，则被认为恶意客户端。

检测恶意客户端：对于每个客户端c_i，计算其LCOF值。若LCOF$(i)>\theta$，则将其标记为恶意客户端，并加入到恶意客户端集合$\mathcal{M}$中。

$$\mathcal{M}=\left\{c_{\mathcal{M}}\middle|c_{\mathcal{M}}=c_i,c_i\in C,\text{LCOF}(i)>\theta\right\} \tag{8}$$

完整的恶意客户端识别算法见算法1。

算法1. 恶意客户端识别算法

输入：类别集合Y、参数集合F、邻居数k、阈值θ

输出：恶意客户端集合$\mathcal{M}$

1. 初始化恶意客户端集合$\mathcal{M}\leftarrow\varnothing$；
2. FOR$\{i=1,2,\cdots,n\}$｛
3. 计算客户端c_i的参数贡献值$\phi_i^{(y)}(l)$；
4. 构建参数贡献集合Φ_i；
5. ｝
6. 构建参数贡献矩阵$\boldsymbol{\Gamma}_{tx}$；
7. FOR$\{i=1,2,\cdots,n\}$｛
8. FOR$\{i=1,2,\cdots,n\}$｛
9. 计算与其他客户端的贡献差异D_{ij}；
10. ｝
11. 获取k个最近邻$C_k(i)$；
12. 计算局部贡献密度$\rho(i)$；
13. ｝
14. FOR$\{j=1,2,\cdots,n\}$｛
15. 计算局部贡献异常因子LCOF(i)；
16. IF LCOF$(i)>\theta$｛
17. $\mathcal{M}\leftarrow\mathcal{M}\cup\{c_i\}$；
18. ｝
19. ｝
20. RETURN $\mathcal{M}$；

(2)动态裁剪算法

由于恶意更新内部参数的异常贡献度存在异质性，准确量化其恶意程度成为关键挑战。为此，本文通过LCOF值对参数恶意性进行量化，并据此针对性地裁剪恶意模型中具有高恶意影响的参数。

具体来说，首先计算良性客户端集合$\mathcal{B}=\left\{c_{\mathcal{B}}\middle|c_{\mathcal{B}}=c_i,c_i\in C,c_i\notin\mathcal{M}\right\}$的中位数$med$和中位数绝对偏差$\|med\|$：

$$med=\text{Median}\left(\left\{w_{t,j}^{(i)}\middle|c_i\in\mathcal{B}\right\}\right) \tag{9}$$

$$\|med\|=\text{Median}\left(\left|w_{t,j}^{(i)}-med\right|\middle|c_i\in\mathcal{B}\right) \tag{10}$$

其中，$w_{t,j}^{(i)}$为训练轮次t时客户端c_i的第j个模型参数，Median$(\cdot)$为中值函数。

利用每个恶意客户端的LCOF值，量化其恶意程度α_i，作为裁剪力度的依据：

$$\alpha_i=\min\left(\frac{\text{LCOF}(i)-\theta}{\text{LCOF}_{\max}-\theta},1\right) \tag{11}$$

其中，$\text{LCOF}_{\max}=\max\limits_{c_i\in\mathcal{M}}\left(\text{LCOF}(i)\right)$，该函数确保$\alpha_i$在$[0,1]$范围内，表示恶意程度的相对强度。

计算每个恶意客户端的参数$w_{t,j}^{(i)}$参数相对偏差$\gamma_{i,j}$，计算公式为：

$$\gamma_{i,j}=\min\left(\frac{\left|w_{t,j}^{(i)}-med\right|}{k\times\|med\|},1\right) \tag{12}$$

其中，k是一个超参数，$\gamma_{i,j}$用于衡量参数$w_{t,j}^{(i)}$与良性客户端集合med的偏离程度，值越大表示异常程度越高。

根据α_i和$\gamma_{i,j}$，计算裁剪因子$\eta_{i,j}$：

$$\eta_{i,j}=(1-\alpha_i)+\alpha_i\times\left(1-\gamma_{i,j}\right) \tag{13}$$

对恶意客户端参数$w_{t,j}^{(i)}$应用裁剪：

$$\hat{w}_{t,j}^{(i)}=\begin{cases}w_{t,j}^{(i)}\times\min\left(1,\eta_{i,j}\right),\left|w_{t,j}^{(i)}-\|med\|\right|\leqslant k\times med\\ med+k\times\|med\|\times\text{sign}\left(w_{t,j}^{(i)}-med\right),\text{else}\end{cases} \tag{14}$$

对恶意模型处于正常范围内的参数应用裁剪因子$\eta_{i,j}$缩放参数$w_{t,j}^{(i)}$。而异常范围内的参数裁剪至$med+k\times\|med\|\times\text{sign}\left(w_{t,j}^{(i)}-med\right)$。

基于上述内容，完整动态裁剪算法见算法2。

算法2.　动态裁剪算法

输入：模型参数$w_{t,j}^{(i)}$，恶意客户端集合$\mathcal{M}$，良性客户端集合$\mathcal{B}$，超参数k，阈值θ

输出：裁剪后的恶意客户端模型参数$\hat{w}_{t,j}^{(i)}$

1. FOR$\{j=1,2,\cdots,m\}$\{
2.　　计算良性客户端的参数中位数med；
3.　　计算中位数绝对偏差$\|med\|$；
4.　\}
5. 计算$\mathrm{LCOF}_{\max}$；
6. FOR$\{c_i\in m\}$\{
7.　　计算恶意程度α_i；
8.　　FOR$\{j=1,2,\cdots,m\}$\{
9.　　计算参数相对偏差$\gamma_{i,j}$；
10.　　计算裁剪因子$\eta_{i,j}$；
11.　　IF 参数在正常范围内 \{
12.　　$\hat{w}_{t,j}^{(i)}\leftarrow w_{t,j}^{(i)}\times\min\left(1,\eta_{i,j}\right)$；
13.　　\} ELSE \{
14. $\hat{w}_{t,j}^{(i)}\leftarrow med+k\times\|med\|\times\mathrm{sign}\left(w_{t,j}^{(i)}-med\right)$；
15.　　\}
16.　\}
17. \}
18. RETURN$\hat{w}_{t,j}^{(i)}$；

3.4　分析与讨论

(1)动态裁剪算法对模型性能的影响分析

在联邦学习框架下，客户端$C=c_{\mathcal{B}}+c_{\mathcal{M}}$，其中$c_{\mathcal{B}}$为良性客户端，$c_{\mathcal{M}}$为恶意客户端。设第$j$个良性客户端$c_j$上传的模型参数为$w_{\mathcal{B}}^{(j)}$，第$i$个恶意客户端$c_i$上传的模型参数为$w_{\mathcal{M}}^{(i)}$。为抑制恶意客户端对全局模型的负面影响，本文引入动态裁剪算法，对恶意客户端上传的参数施加裁剪因子η_i，使得裁剪后的参数表示为：

$$\hat{w}_{\mathcal{M}}^{(i)}=\eta_i\cdot w_{\mathcal{M}}^{(i)} \tag{15}$$

服务端聚合所有客户端上传的模型参数后，裁剪处理后的全局模型参数G'更新可表示为：

$$G'=\frac{1}{C}\left(\sum_{j=1}^{c_{\mathcal{B}}}w_{\mathcal{B}}^{(j)}+\sum_{i=1}^{c_{\mathcal{M}}}\hat{w}_{\mathcal{M}}^{(i)}\right) \tag{16}$$

在统计意义下，良性客户端参数与恶意客户端参数相互独立，则全局模型参数的期望为：

$$\mathbb{E}[G']=\frac{1}{C}\left(\sum_{j=1}^{c_{\mathcal{B}}}\mathbb{E}\left[w_{\mathcal{B}}^{(j)}\right]+\sum_{i=1}^{c_{\mathcal{M}}}\mathbb{E}\left[\hat{w}_{\mathcal{M}}^{(i)}\right]\right) \tag{17}$$

定义良性客户端参数的均值为：

$$\mathbb{E}\left[w_{\mathcal{B}}^{(j)}\right]=\frac{1}{c_{\mathcal{B}}}\sum_{j=1}^{c_{\mathcal{B}}}w_{\mathcal{B}}^{(j)} \tag{18}$$

恶意客户端参数的均值为：

$$\mathbb{E}\left[\hat{w}_{\mathcal{M}}^{(i)}\right]=\frac{1}{c_{\mathcal{M}}}\sum_{i=1}^{c_{\mathcal{M}}}\hat{w}_{\mathcal{M}}^{(i)} \tag{19}$$

则期望可重写为：

$$\mathbb{E}[G']=\frac{c_{\mathcal{B}}}{C}\mathbb{E}\left[w_{\mathcal{B}}^{(j)}\right]+\frac{c_{\mathcal{M}}}{C}\mathbb{E}\left[\hat{w}_{\mathcal{M}}^{(i)}\right] \tag{20}$$

通过设计动态裁剪因子η_i满足$\eta_i<\alpha<1$，从而有效降低恶意客户端上传参数的期望贡献：

$$\mathbb{E}\left[\hat{w}_{\mathcal{M}}^{(i)}\right]<\alpha\mathbb{E}\left[w_{\mathcal{M}}^{(i)}\right] \tag{21}$$

因此，动态裁剪算法能够有效抑制恶意客户端参数对全局模型的负面影响。

(2)动态裁剪算法对梯度方向的影响分析

假设全局模型的损失函数为$L(G)$，传统的全局模型参数更新遵循标准梯度下降规则，即：

$$G_{t+1}=G_t-\zeta\nabla L(G_t) \tag{22}$$

其中，$\zeta>0$为学习率。引入动态裁剪算法后，恶意参数被裁剪，形成新的全局参数：

$$G'=\frac{1}{C}\left(\sum_{j=1}^{c_{\mathcal{B}}}w_{\mathcal{B}}^{(j)}+\sum_{i=1}^{c_{\mathcal{M}}}\hat{w}_{\mathcal{M}}^{(i)}\right) \tag{23}$$

在接下来的训练轮次中，中央服务器基于G'计算梯度，得到新的梯度$\nabla L(G')$。由于动态裁剪设计保证了良性客户端的参数$w_{\mathcal{B}}^{(j)}$不受影响，恶意参数$w_{\mathcal{M}}^{(i)}$的影响被削弱。所以更新后的全局参数G'更加接近良性客户端主导下的模型参数记作：

$$G_{\mathcal{B}}=\frac{1}{c_{\mathcal{B}}}\sum_{j=1}^{c_{\mathcal{B}}}w_{\mathcal{B}}^{(j)} \tag{24}$$

在此基础上，若损失函数$L(G)$满足L-Lipschitz光滑条件，即其梯度关于模型参数是Lipschitz连续的，则有：

$$\nabla L\left(\mathrm{G}'\right)-\nabla L(G_{\mathcal{B}})<L\cdot\left\|\mathrm{G}'-G_{\mathcal{B}}\right\|<\epsilon_1 \tag{25}$$

其中，ϵ_1为可控的小概率误差项，反映了由于恶意扰动残留引起的梯度偏差上限。该不等式表明，在满足损失函数连续性和平滑性条件下，基于裁剪后模型参数G'计算得到的梯度方向，与理想情况下(即无攻击、仅由良性客户端贡献得到模型参数$G_{\mathcal{B}}$)计算的梯度方向之间的偏差可被有效控制在较小范围内。

(3)Non-IID下恶意客户端检测精度分析

在Non-IID下，各良性客户端的数据分布不同，因此参数贡献集合Φ_i会存在差异，但对于任意$c_{\mathcal{B}}$，存在一个可学习的分布$P_{\mathcal{B}}(\Phi_i)$，使得：

$$\mathbb{P}(\Phi_i\sim P_{\mathcal{B}})\geqslant 1-\epsilon_1 \tag{26}$$

其中，ϵ_1是小概率误差项。恶意客户端$c_{\mathcal{M}}$的目标是

通过投毒来影响全局模型，因此它们的参数贡献集合Φ_i会偏离良性客户端的统计分布，形成分布$P_{\mathcal{M}}$，且：

$$D_{\mathrm{KL}}\left(P_{\mathcal{M}} \| P_{\mathcal{B}}\right) \gg 0 \tag{27}$$

即恶意客户端的贡献分布与良性客户端的贡献分布在KL散度意义下具有显著区分度。此时，SHAP参数贡献矩阵Γ_{tx}反映出的贡献差异满足：

$$D_{ij}=\sum_{y=1}^{\|Y\|}\| \Phi_i^{(y)}-\Phi_j^{(y)}\|^2 \tag{28}$$

对所有良性客户端c_i和正常邻居c_j有：

$$\mathbb{P}(D_{ij}>\tau)\geqslant 1-\epsilon_2 \tag{29}$$

其中，τ是良性客户端贡献波动范围的上界，ϵ_2为小概率误差项。

LCOF量化了c_i在其邻域中的相对贡献密度。虽然良性客户端的贡献密度存在一定波动，但在局部邻域内仍然满足：

$$\mathbb{P}\left(1-\gamma\leqslant \mathrm{LCOF}(i)\leqslant 1+\gamma\right)\geqslant 1-\epsilon_3, \forall c_i\in\mathcal{B} \tag{30}$$

由于恶意客户端的贡献密度偏离正常统计分布，其LCOF指标远离正常范围，即：

$$\mathbb{P}(\mathrm{LCOF}(m)>\theta)\geqslant 1-\epsilon_4, \forall c_i\in\mathcal{M} \tag{31}$$

选取合适的检测阈值θ，使得良性客户端的贡献波动范围与恶意客户端的异常贡献密度之间形成明显的区分界限，使得：

$$1+\gamma<\theta<\delta \tag{32}$$

则恶意客户端与良性客户端可以被区分的概率至少为：

$$\begin{aligned}&\mathbb{P}\left(\mathrm{LCOF}(i)\leqslant\theta \,|\, c_i\in\mathcal{B}\right)\geqslant 1-\epsilon_3\\&\mathbb{P}(\mathrm{LCOF}(i)>\theta \,|\, c_i\in\mathcal{M})\geqslant 1-\epsilon_4\end{aligned} \tag{33}$$

ϵ_3、ϵ_4为小概率误差项。因此，整体检测精度可以保证至少为：

$$1-(\epsilon_3+\epsilon_4) \tag{34}$$

当ϵ_3和ϵ_4足够小时，恶意客户端检测算法在Non-IID情况下仍能保持较高的检测精度。

4 实验与分析

本节介绍了CADC的部署环境，并验证了其在防御投毒攻击方面的有效性，同时对比分析了与现有防御方法的优劣。实验采用PyTorch框架实现联邦学习的分布式训练与客户端管理，具体的实验环境和超参数设置见表1。

表1 实验环境

软硬件配置	版本
CPU	Intel i5-12600KF
GPU	NVIDIA GeForce RTX 4060 Ti
内存	32 G
操作系统	Windows 10
Pytorch	2.1.0

4.1 数据集及模型架构

本文在三个数据集上评估了CADC的性能，每个数据集分为训练集和测试集，比例为0.8∶0.2，并设置了与数据集相对应的三种不同模型架构。

MNIST数据集①：包含6万张0至9的手写数字图像。针对该数据集，本文采用包含两个卷积层的卷积神经网络模型。

KDDCup数据集②：包含125,973条网络流量记录，涵盖正常流量和多种攻击类型。本文为该数据集构建了由多层全连接层和Dropout层组成的深度神经网络模型。

Amazon数据集③：包含约34,686,770条用户评论，涵盖电子产品、图书、服装、家居用品等多个品类。针对该数据集，本文采用基于双向长短期记忆网络的深度学习模型。

在IID条件下，数据在所有用户之间均匀分配，而在Non-IID条件下，数据分布不均衡，每个用户的数据主要集中于少数几个类别。在参数训练方面，针对MNIST和KDDCup，本文设定了300次迭代、批次大小为50以及学习率为0.01。由于Amazon具有更多的特征且类别样本较少，因此将迭代次数、批次大小和学习率依次调整为60、10和0.005。

4.2 攻击设置

本文评估了四种常见的联邦学习攻击方式，包括标签翻转攻击、局部模型投毒攻击、缩放攻击和虚假客户端攻击，涵盖了从数据层、更新层到系统层的典型攻击路径。

标签翻转攻击[21]：标签翻转攻击是一种典型的数据投毒策略，攻击者通过篡改训练样本的标签信息，在本地训练过程中引导模型学习错误的判别规则，从而影响全局模型性能。该攻击易于实施且隐蔽性强，广泛存在于无中心化监管的数据场景中。

局部模型投毒攻击[25]：局部模型投毒攻击直接

① http://yann.lecun.com/exdb/mnist/
② http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.
③ https://jmcauley.ucsd.edu/data/amazon/

操控客户端的模型参数或其更新结果，使其在上传阶段对全局模型训练产生系统性干扰。此类攻击属于更新层面攻击，往往通过多轮次的扰动实现对模型收敛过程的破坏，具备较强的欺骗性和持久性。

缩放攻击[26]：在缩放攻击中，攻击者通过对局部更新向量进行缩放，使其扰动幅度控制在防御机制可容忍的范围之内，从而绕过聚合层的异常检测。此类攻击尤其在基于模长或相似度过滤机制下表现出较强的隐蔽性。

虚假客户端攻击[27]：虚假客户端攻击通过伪造多个客户端身份，协同上传恶意更新，实现对全局模型更新方向的系统性操控。该攻击方式利用身份认证机制的漏洞放大攻击影响，具备高扩展性与高破坏性，能在未显著增加单一客户端扰动强度的前提下，显著影响聚合结果，降低模型整体性能。

以上攻击类型涵盖了不同攻击目标（精度下降或方向偏移）、干预层级（数据/更新/系统）、扰动方式（显性/隐性）、执行策略（单点/协同）等特征，具有良好的代表性，能够有效支撑对防御方法在多样化攻击环境中的稳定性与适应性的综合评估。

标签翻转攻击和局部模型投毒攻击依赖恶意客户端对训练数据或模型参数的直接操控，其破坏效果通常与恶意客户端数量成正比。因此，本文将这两类攻击的恶意客户端比例设定为20%。缩放攻击和虚假客户端攻击无需依赖大量恶意客户端便能实现破坏效果，因此实验中将其恶意客户端比例设置为15%。实验相关参数的取值如表2所示。

表2　实验参数取值

参数名称	取值
恶意客户端比例	15%，20%
k个最近邻	5
LCOF检测阈值θ	1.5

4.3　比较的防御方法

本文对比的防御方法包括基于欧氏距离的Krum聚合方法[29]、基于得分排名的Zeno方法[30]、通过模型更新贡献剪裁的Clipping方法[33]、结合Shapley加性解释值和支持向量机的SAHP-SVM方法[48]、基于信任引导的FLTrust方法[34]、基于全局模型参数相似性的FedRoLa方法[37]，以及基于Fisher差异聚类的FDCR方法[38]。此外，本文构建了一个以平均值作为聚合规则的联邦学习基准模型，该模型未采用任何防御方法，用于与其他防御方法进行对比分析。

4.4　评价指标

本文采用多种评价指标来衡量防御方法对恶意客户端的检测效果及其对全局模型的影响。这些指标基于以下四种基本分类结果：True Positive（TP）：恶意客户端被成功检测为恶意客户端；False Positive（FP）：良性客户端被误判为恶意客户端；True Negative（TN）：良性客户端被正确识别为良性客户端；False Negative（FN）：恶意客户端未被检测出，而被错误分类为良性客户端。

恶意客户端识别准确率（Detection Accuracy Rate，DAR）：衡量防御方法在所有样本（包括良性客户端和恶意客户端）中成功检测到恶意客户端的比例，定义如下：

$$DAR=\frac{TP}{TP+FP+TN+FN} \tag{35}$$

恶意客户端识别率（True Positive Rate，TPR）：衡量防御方法对恶意客户端的检测能力，定义为成功检测出的恶意客户端占所有恶意客户端的比例，计算公式如下：

$$TPR=\frac{TP}{TP+FN} \tag{36}$$

误报率（False Positive Rate，FPR）：表示被误判为恶意客户端的良性客户端占总良性客户端的比例，衡量防御方法对良性客户端的误伤情况，计算公式如下：

$$FPR=\frac{FP}{TN+FP} \tag{37}$$

F1-score：综合考虑TPR和FPR，以平衡检测效果。

模型准确性（Accuracy，Acc）：衡量全局模型在测试集上的整体分类准确率，即模型正确预测的样本数占总样本数的比例。

4.5　实验评估

对CADC的评估主要从以下六个方面开展：恶意客户端的识别准确率、不同攻击下恶意客户端数量对防御方法的影响、Non-IID对防御效果的影响、全局模型的准确率、消融实验以及计算开销。

（1）恶意客户端的识别准确率

本节比较了CADC与现有防御方法在不同攻击场景和数据集下的恶意客户端识别准确率。如图2所示，在多种投毒攻击中，CADC的DAR在各数据集上均优于当前主流方法。

Krum依赖局部距离筛选，易在攻击者利用缩放降低恶意更新与正常更新距离时误判异常。Zeno

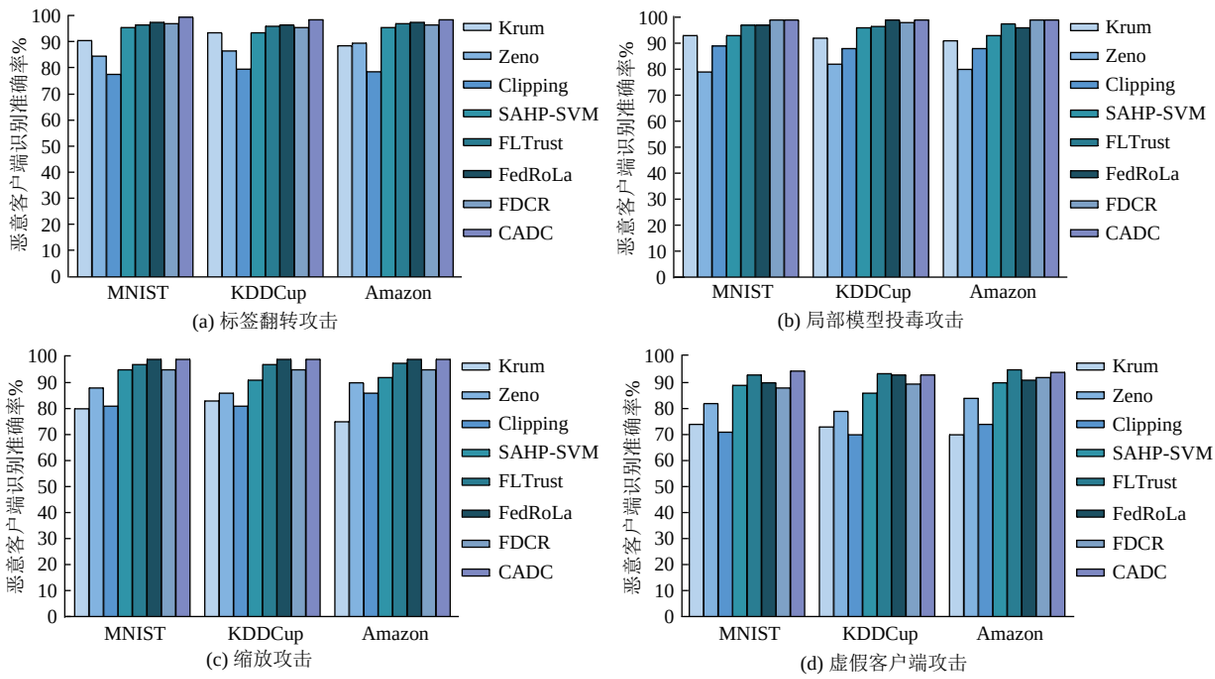


图2 恶意客户端识别准确率

侧重评估更新的整体贡献，当攻击者对局部模型投毒进行微调，使其在整体下降趋势上与正常更新保持一致时，Zeno难以识别这类隐蔽异常。Clipping方法仅限制更新幅度，未考虑更新方向及其统计特征，导致在缩放攻击下异常更新仍能规避检测。SAHP-SVM借助Shapley值与SVM分类提升识别能力，但其理论假设对渐进式攻击的敏感性不足，难以及时捕捉持续微弱的扰动。FLTrust依赖根数据集与更新方向的一致性进行防御，但在面对伪造更新的虚假客户端时，基于方向一致性的前提难以成立，从而削弱识别能力。FedRoLa通过逐层聚合与动态层选择检测异常，但由于不同层在模型中的重要性存在差异，攻击者若针对更新幅度较小或本身波动较大的层实施操控，易形成检测盲区。FDCR方法基于Fisher信息矩阵评估参数重要性，能有效识别由标签翻转和局部投毒引起的局部异常更新，但其未考虑客户端整体贡献度，且缩放攻击不会改变更新方向，易规避此类检测。

相比之下，CADC在多个攻击场景中展现出更强的鲁棒性。对于标签翻转攻击，该方法可以通过参数贡献矩阵与差异度量，捕捉特定类别参数贡献的异常变化，恶意客户端在整体贡献分布上表现出偏离。对于局部模型投毒攻击，CADC在多类别特征空间中评估参数贡献差异，即便攻击较为隐蔽，随着迭代进行，恶意客户端的贡献分布仍会逐步偏离良性客户端，并在局部异常因子的计算中体现为异常低的局部贡献密度。面对缩放攻击，CADC可利用局部贡献密度与异常因子识别正常客户端更新的聚类结构，而缩放后的恶意更新尽管模长受限，其在局部分布中依然表现出稀疏、偏离整体的特征。对于虚假客户端攻击，其伪造的更新在参数贡献上的不一致性会被LCOF放大，从而识别出恶意客户端。

(2)恶意客户端比例对防御方法的影响

本节讨论了恶意客户端比例对不同防御方法检测准确率的影响。由图3～图5所可知，随着恶意客户端数量的增加，除CADC方法外，其余方法的检测准确率均明显下降。

Krum方法通过欧氏距离识别异常，但在恶意客户端比例较高时，恶意更新与正常更新距离缩小，导致距离度量失效，检测效果下降。Zeno基于得分排名机制，面对高比例恶意更新时，整体贡献分布被扰动，决策边界模糊，识别准确率降低。FLTrust依赖更新方向一致性，当恶意客户端增多，其更新干扰整体方向，削弱区分能力。FedRoLa依赖全局参数相似性，恶意更新比例上升会污染正常更新的相似性特征，防御能力减弱。FDCR在低至中等恶意比例下表现稳定，但恶意更新数量增多时易形成一致聚类，引发误判。相比之下，CADC结合SHAP值与局部异常因子，量化

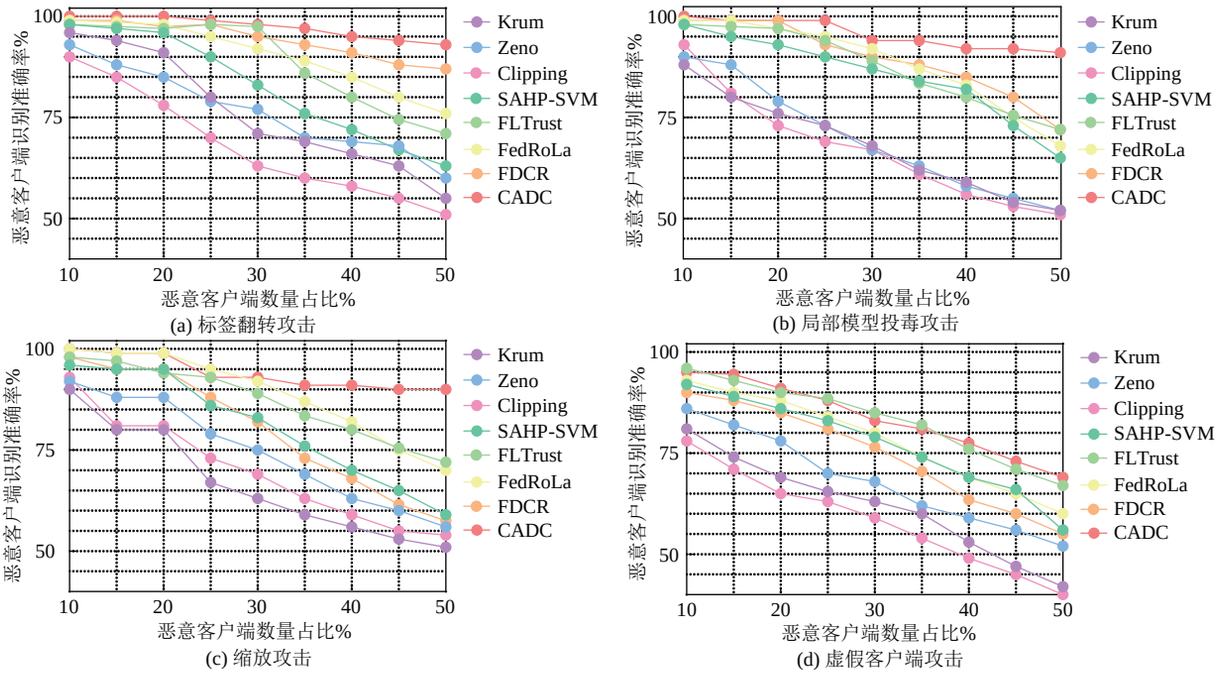


图3　MNIST数据集上恶意客户端识别率

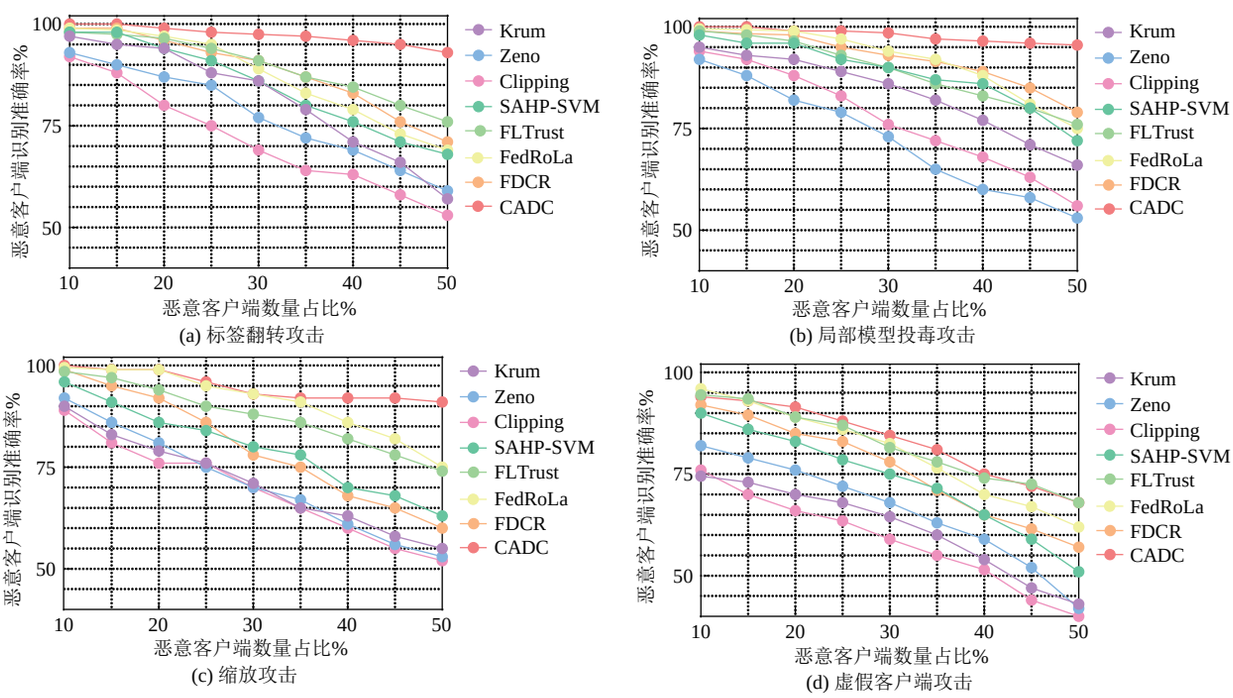


图4　KDDCup数据集上恶意客户端识别率

客户端参数对模型预测的贡献分布。正常更新基于真实数据，贡献模式稳定且聚集，而恶意更新虽部分参数相似，但在特定维度的贡献差异显著，暴露异常行为。

（3）Non-IID对识别准确率的影响

本节分析了不同Non-IID程度下，随着集中参数β的增大（即β越大，客户端之间的数据分布越趋一致[7]），各防御方法识别准确率变化如图6至图8所示。

在Non-IID条件下，客户端因数据类别不均，正常更新之间差异增大，使模型更新在特征空间中呈现较高离散性，影响基于距离或统计贡献的防御方

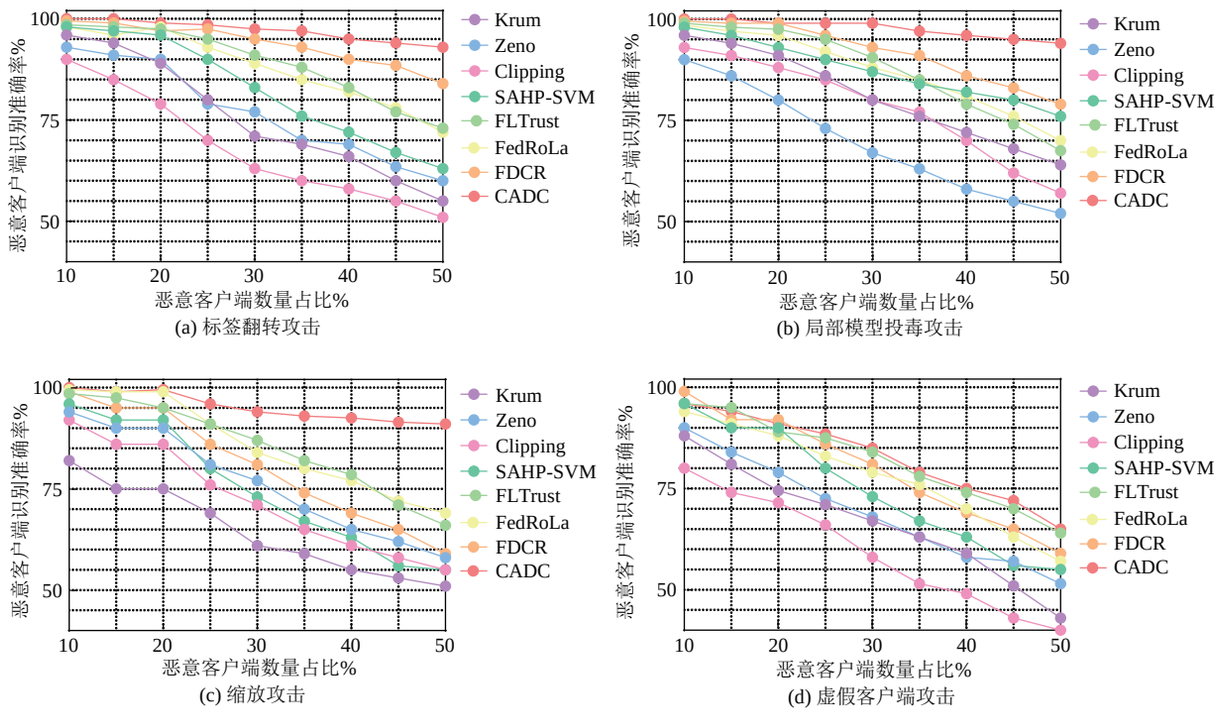


图5 Amazon数据集上恶意客户端识别率

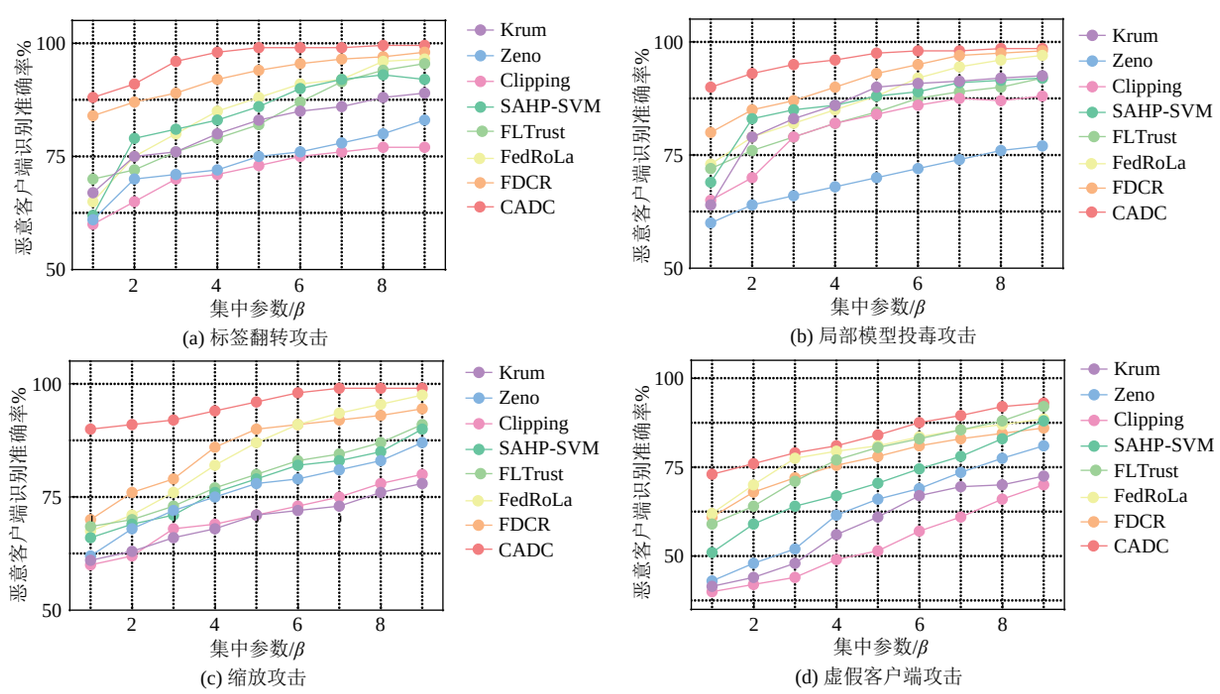


图6 MNIST数据集Non-IID程度对恶意客户端识别的影响

法的判别能力。这种差异性容易引发“伪异常”现象，掩盖恶意更新所造成的实际扰动。CADC方法通过结合局部贡献密度和局部异常因子建模参数贡献的局部结构，即使在β较低的场景中，也能识别出恶意更新在特定维度上的离群行为。尽管正常客户端在参数贡献空间中差异显著，但恶意客户端的局部密度更稀疏，异常因子得分更高，仍可被有效区分。在β较高、数据分布趋于一致时，正常客户端的贡献模式收敛，局部结构紧密，恶意更新引起的偏离更加明显，识别效果进一步提升。

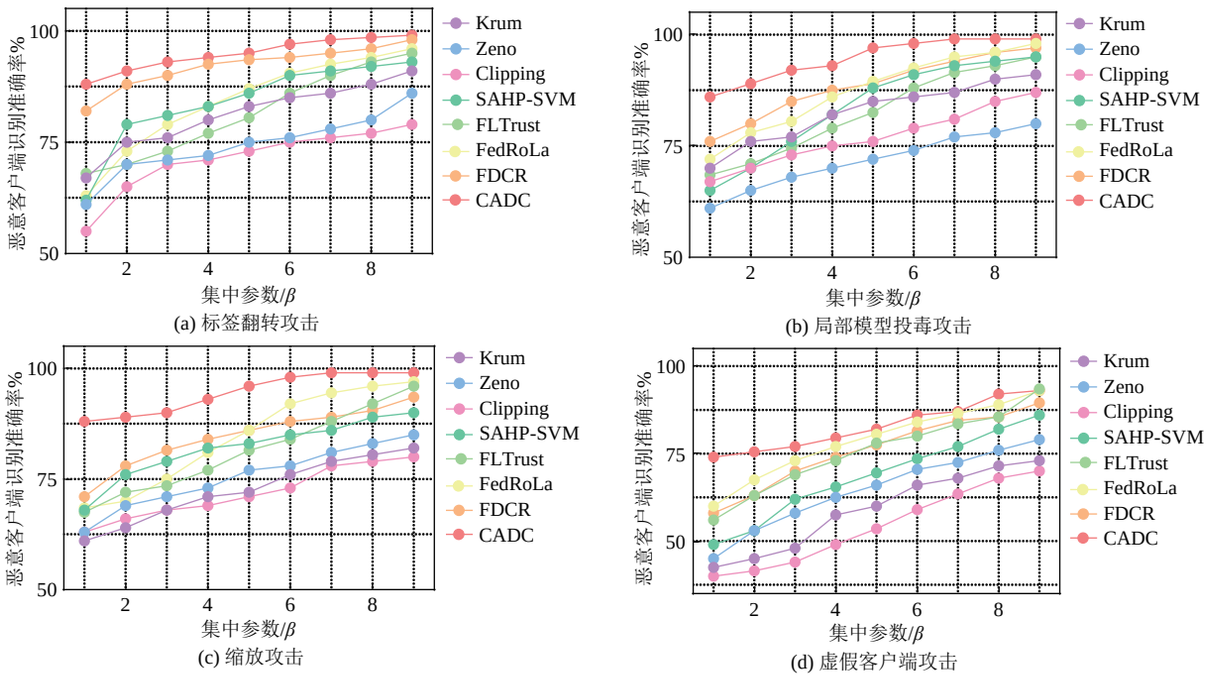


图7　KDDCup数据集Non-IID程度对恶意客户端识别的影响

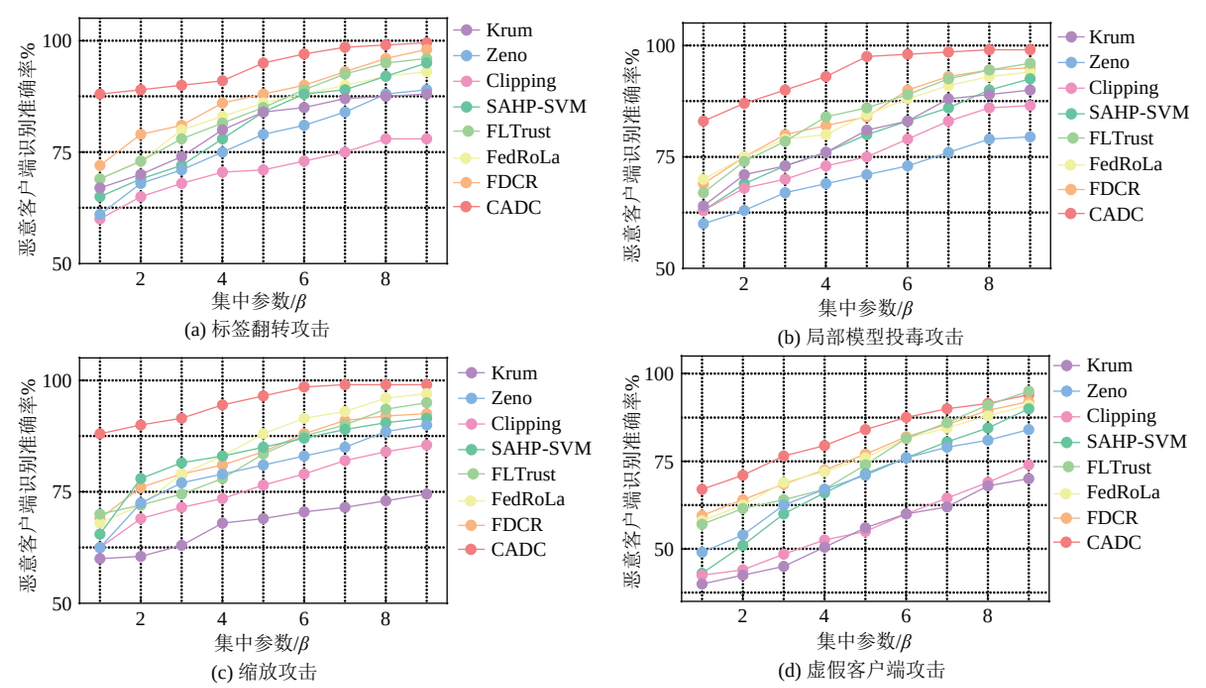


图8　Amazon数据集Non-IID程度对恶意客户端识别的影响

(4)全局模型的准确率

如表3所示，CADC方法在各数据集与不同攻击类型下均实现较高全局模型准确率。CADC方法通过引入LCOF值量化各参数的恶意程度，实现对恶意程度最大参数的裁剪，减轻裁剪操作对模型收敛的负面影响，从而提高全局模型准确率。

(5)消融实验

为验证SHAP与SV在揭示联邦学习客户端参数异常贡献上的差异，本节在Amazon数据集上开展了实验。实验将数据划分为10个客户端，其中前5个客户端被设定为恶意客户端，对部分文本内容进行标签翻转来模拟攻击行为。我们分别计算各客

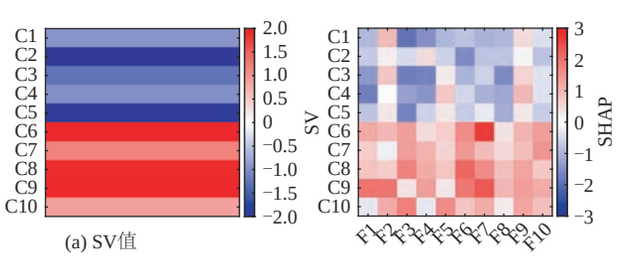
表3 不同攻击和不同检测方法下的全局模型的准确率

数据集	防御方法	无攻击	标签翻转攻击	局部模型投毒攻击	缩放攻击	虚假客户端攻击
MNIST	Krum	89.05	81.25	82.05	74.65	67.47
	Zeno	89.05	78.65	75.30	81.36	69.39
	Clipping	89.05	77.56	80.04	75.61	64.73
	SAHP-SVM	89.05	80.12	80.16	82.04	74.64
	FLTrust	89.05	82.47	82.43	82.73	80.16
	FedRoLa	89.05	83.54	82.72	79.65	78.14
	FDCR	89.05	78.50	79.64	77.94	66.50
	无防御	89.05	12.36	13.05	9.12	8.57
	CADC	**89.05**	**87.86**	**87.65**	**88.06**	**82.14**
KDDCup	Krum	73.10	69.23	68.65	62.16	54.31
	Zeno	73.10	62.41	60.12	67.31	57.49
	Clipping	73.10	60.57	62.85	63.96	53.67
	SAHP-SVM	73.10	70.05	69.34	64.96	60.59
	FLTrust	73.10	69.46	70.14	64.31	62.19
	FedRoLa	73.10	68.50	68.87	65.59	61.13
	FDCR	73.10	70.42	71.23	67.54	60.64
	无防御	73.10	10.85	11.37	8.63	8.64
	CADC	**73.10**	**71.42**	**72.69**	**72.91**	**65.46**
Amazon	Krum	81.32	76.81	74.31	69.31	58.37
	Zeno	81.32	72.64	67.48	73.93	63.17
	Clipping	81.32	73.87	72.64	69.94	59.43
	SAHP-SVM	81.32	76.87	75.49	74.19	65.46
	FLTrust	81.32	77.16	77.16	73.31	70.43
	FedRoLa	81.32	75.43	74.78	71.83	69.37
	FDCR	81.32	79.15	78.51	70.53	68.43
	无防御	81.32	11.32	10.54	8.93	8.43
	CADC	**81.32**	**80.98**	**80.13**	**79.84**	**78.12**

户端的SV值和SHAP值，对比各客户端在参数层面的贡献情况。

实验结果如图9所示，SV方法只能以单一数值量化每个客户端的整体贡献，而SHAP值揭示出恶意客户端在参数上的异常贡献。这表明，SHAP方法不仅能够识别出恶意客户端，还能定位攻击影响的特定参数，从而为异常检测和后续防御策略提供更具针对性的解释依据。

为了验证CADC的有效性，本文在MNIST数据集上，针对多种攻击类型，做以下两种消融设置：(1)去除LCOF指标。在该设置中，预先标记良性客户端，并利用其SHAP值统计数据构建良性基准，采用中位数绝对偏差作为衡量指标。当检测客户端的SHAP统计量偏离该基准超过2倍中位数绝对偏差时，将其判定为异常；(2)采用无权重裁剪方法。在此设置下，直接剔除所有被判定为恶意的客户端上传的模型参数，而不对其中可能存在的良性参数进行保留。



(a) SV值 (b) SHAP值

图9 热力值对比图

如图10所示，仅依赖SHAP值进行检测时，识别准确率不及完整的CADC方法，尤其在局部模型投毒攻击和虚假客户端攻击下表现不佳，说明LCOF指标在进一步识别异常客户端方面起到了关键作用。

如表4所示，在四种攻击方式下，无权重裁剪方法的全局模型准确率低于动态裁剪算法。动态裁剪算法通过保留恶意模型的有益参数，提升了全局模型准确性。

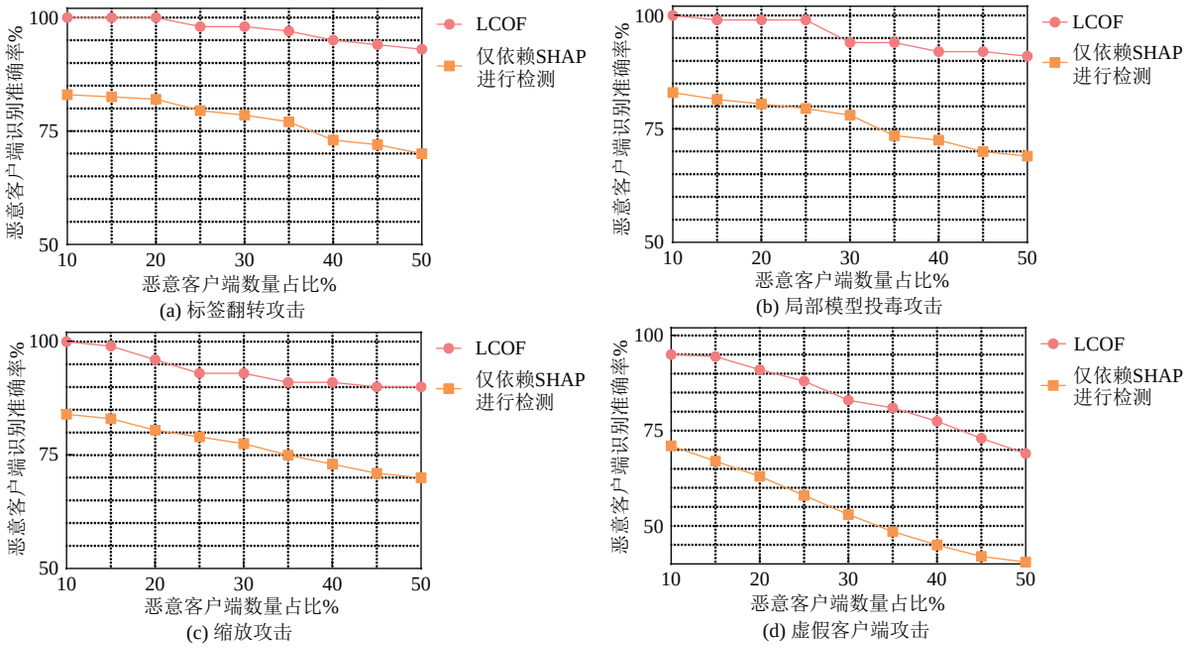


图10　消融设置下恶意客户端识别准确率

表4　消融设置下全局模型的准确率

数据集	方法	标签翻转攻击	局部模型投毒攻击	缩放攻击	虚假客户端攻击
MNIST	无权重裁剪	78.65%	75.30%	81.36%	60.43
	CADC	**87.86%**	**87.65%**	**88.06%**	**82.14**

(6)关键参数

LCOF检测阈值θ决定了客户端被判定为异常的标准。当θ过低时，可能会导致良性客户端被误判。当θ过高时，则可能导致恶意客户端未被检测到。为了寻找最佳阈值θ，使TPR和FPR达到最佳平衡，本文在MNIST数据集上，针对标签翻转攻击，固定$k=5$，调整θ取不同值$\{0.5, 1.0, 1.5, 2.0\}$，并评估不同θ对TPR、FPR、F1-score以及全局模型准确率的影响。实验结果如表5所示。

表5　LCOF检测阈值θ选择实验结果

θ	TPR	FPR	F1-score	全局模型准确率
0.5	100%	20.35%	91.05%	84.21%
1.0	98.72%	10.42%	93.21%	86.73%
1.5	**97.56%**	**5.89%**	**95.32%**	**87.86%**
2.0	89.24%	2.31%	91.47%	86.95%

当θ设定较低(如0.5)时，尽管TPR达到了100%，但FPR也高达20.35%，导致大量良性客户端被误判，全局模型准确率下降至84.21%。相反，当θ设定较高(如2.0)时，FPR降低至2.31%，但TPR也下降至89.24%，部分恶意客户端未被成功检测。在实验中，当θ取1.5时，TPR维持在97.56%，FPR降至5.89%，F1-score最高达95.32%，全局模型准确率达到87.86%。结果表明$\theta=1.5$能够有效平衡恶意客户端的检测能力和误报率。

在LCOF方法中，最近邻数k影响恶意客户端的检测精度。因此，本文固定$\theta=1.5$，调整k取不同值，评估不同k对TPR、FPR、F1-score以及全局模型准确率的影响。实验结果如表6所示。

表6　最近邻数k选择实验结果

k	TPR	FPR	F1-score	全局模型准确率
3	98.93%	8.21%	94.12%	86.32%
5	**97.56%**	**5.89%**	**95.32%**	**87.86%**
10	96.24%	4.31%	94.76%	84.14%

当k较小时，局部邻域数据较少，易受个体差异干扰，导致检测过于敏感，使得TPR虽然高达98.93%，但FPR也升至8.21%，进而影响全局模型的准确率。而当k较大时，局部密度估计趋于平滑，部分异常参数可能被局部平均化，导致TPR下降，恶意客户端的漏检问题也随之出现。实验结果显示，当k取5时，能够在保持较高TPR(97.56%)的

同时，将FPR控制在5.89%，使得F1-score最高达到95.32%，且全局模型准确率也达到了87.86%。因此，在本实验中，$k=5$是一个较优的选择，既能捕捉局部异常，又避免了误判良性客户端，实现检测性能与全局模型效果的平衡。

(7)计算开销

CADC方法需要为所有参数组合计算SHAP值以确定参数属性，这带来了较大的计算开销。为此，本文在恶意模型检测中，SHAP值的原始计算复杂度为$O(n\times f\times 2^f)$(f为参数数量)。为降低计算负担，本文采用基于蒙特卡洛采样的近似算法[49-50]，根据各参数的方差调整采样步数m，使得在估计SHAP值收敛时提前停止采样，从而将复杂度降至$O(n\times f\times m)$。在构建参数贡献矩阵及计算贡献差异的过程，复杂度为$O(n^2\times f)$。为加速这一过程，本文采用了矩阵分块与并行处理技术[51]，根据矩阵中各区域的计算量和数据规模，将贡献矩阵划分为多个子块，使得每个子块的计算负担均衡，将这些子块分配到多个CPU线程中，实现多线程并行处理，从而在不同子块间同时计算贡献差异。

基于上述理论分析，本文在不含客户端本地训练与通信延迟的单轮聚合阶段对各防御方法的服务器端防御模块计算时间进行了实验比较。结果如见表7所示。

表7 各方法的服务器端防御计算时间

防御方法	单轮防御计算时间(s)		
	MNIST	KDDCup	Amazon
Krum	6.53	7.21	3.64
Zeno	7.04	8.32	4.03
Clipping	6.02	6.54	3.26
SAHP-SVM	22.53	24.67	12.91
FLTrust	7.19	7.94	7.33
FedRoLa	18.23	19.51	5.13
FDCR	15.27	16.34	4.57
无防御	**5.54**	**6.75**	**3.34**
CADC	**21.82**	**23.12**	**5.82**

在实际联邦学习场景中，客户端采用周期性提交更新方式(详见表8)，使得服务器端防御任务在整个训练过程中得到分摊。

完整训练过程总运行时间的数据如表9所示。

由表7、表9实验数据可以看出，在单轮聚合阶段，CADC方法的服务器端防御计算时间最高，主要由于CADC需要为所有参数组合计算SHAP值，从而导致较大的计算开销。尽管CADC方法使总训练时间相较于无防御方法略有增加，但整体延时增幅并不显著，这表明其计算开销对整体训练周期的影响是可控的。

表8 模拟联邦学习场景客户端提交设置

数据集	总客户端数	每轮提交比例	每轮间隔时间(s)
MNIST	100	20%	20
KDDCup	120	20%	20
Amazon	80	20%	10

表9 各方法完整训练过程总运行时间

防御方法	总训练时间(min)		
	MNIST	KDDCup	Amazon
Krum	145	162	36
Zeno	158	175	38
Clipping	141	162	35
SAHP-SVM	226	248	98
FLTrust	154	167	38
FedRoLa	155	170	37
FDCR	158	171	38
无防御	**132**	**150**	**32**
CADC	**164**	**178**	**39**

为进一步验证各防御方法在实际部署时对系统资源的占用情况，本文记录了服务器在防御模块运行期间的平均资源占用指标，包括CPU占用率及GPU占用率。实验结果如表10所示。

表10 系统资源占用情况(平均值)

防御方法	CPU占用率(%)	GPU占用率(%)
Krum	55	40
Zeno	58	42
Clipping	54	38
SAHP-SVM	75	60
FLTrust	52	40
FedRoLa	70	55
FDCR	65	50
无防御	**50**	**35**
CADC	**79**	**63**

如表10所示，CADC方法的CPU和GPU占用率均高于其他防御方法，这反映出其在计算SHAP值时对系统资源的较高需求。尽管如此，考虑到联邦学习中央服务器通常具备充足的计算资源，此类计算开销是可接受的。总体来看，CADC方法在保证较高恶意检测精度的同时，其计算开销和资源消耗均在合理范围内，验证了其在安全防护方面的实际应用价值。

4.6 结果讨论

本文在四类典型投毒攻击下开展了系统实

验，涵盖了从数据层、更新层到系统层的典型攻击路径。实验结果表明，所提方法在不同类型和强度的攻击下均展现出较好的鲁棒性与适应性。在恶意客户端识别方面，基于SHAP的参数层贡献分析结合局部异常因子，能够揭示客户端更新的异常模式，有效识别多种攻击策略下的隐蔽行为。在模型聚合阶段，设计的动态裁剪算法结合恶意程度与参数相对偏差进行调控，实现了对攻击扰动的有针对性压制，同时最大限度保留良性更新信息，整体提升了防御的灵活性与准确性。对于更具隐蔽性与复杂性的攻击策略，如触发式后门攻击、多轮微扰协同攻击与动态身份伪装攻击等，本文所提方法也具有一定的理论防御能力。触发式后门攻击虽然在整体预测表现上趋近正常，但在特定输入条件下会引发异常输出，若其训练行为在参数层贡献中产生可测差异，仍会被SHAP贡献矩阵捕捉。对于多轮微扰或协同式攻击，尽管单轮扰动较小，但在聚合贡献矩阵中可通过跨客户端的贡献密度聚合体现其系统性偏移，从而激活局部异常因子机制予以识别。

5　结束语

针对联邦学习中由恶意客户端引发的安全威胁，本文提出了一种融合参数贡献分析与动态裁剪策略的鲁棒防御方法。该方法通过构建基于SHAP值的参数贡献矩阵，实现参数层面的异常检测。在此基础上，引入动态裁剪算法，对模型参数进行差异化裁剪，抑制恶意扰动，同时保留有效信息。实验结果验证了本方法在多种典型投毒攻击场景下均具备良好的防御效果和模型性能保持能力。未来的研究将进一步引入输入激活路径分析、身份追踪机制及跨轮行为一致性建模等技术手段，增强对拟态能力强、具备身份变换策略的逃逸型攻击与伪装式攻击的防御能力，从而推动联邦学习系统在安全性与实用性方面的进一步发展。

致　谢　衷心感谢编辑以及各位审稿专家对本文工作所给予的宝贵建议！

参　考　文　献

[1] Chen Haoyu, Li Wudong, Zhang Honglei, et al. A review of research on fairness in trusted federated learning. Acta Electronica Sinica, 2023, 51(10): 2985-3010 (in Chinese)
(陈颢瑜，李浥东，张洪磊，等.面向可信联邦学习公平性的研究综述.电子学报，2023，51(10):2985-3010)

[2] Zhou X, Ye X, Kevin I, et al. Hierarchical federated learning with social context clustering-based participant selection for internet of medical things applications. IEEE Transactions on Computational Social Systems, 2023, 10(4): 1742-1751

[3] Jiang Weijin, Du Xichen, Jiang Yirong, et al. A swarm intelligence perception algorithm for environmental monitoring based on adaptive federated learning. Acta Electronic Sinica, 2025, 53(03): 821-835 (in Chinese)
(蒋伟进，杜熙晨，蒋意容，等.基于自适应联邦学习的环境监测群智感知算法.电子学报，2025，53(03):821-835)

[4] Gu Yuhao, Bai Yuebin. Research progress on security and privacy of federated learning models. Journal of Software, 2022, 32(06): 2833-2864 (in Chinese)
(顾育豪，白跃彬.联邦学习模型安全与隐私研究进展.软件学报，2022，32(06):2833-2864)

[5] Uddin M P, Xiang Y, Hasan M, et al. A systematic literature review of robust federated learning: issues, solutions, and future research directions. ACM Computing Surveys, 2025, 57(10): 1-62

[6] Zhou X, Liang W, Kevin I, et al. Decentralized P2P federated learning for privacy-preserving and resilient mobile robotic systems. IEEE Wireless Communications, 2023, 30(2): 82-89

[7] Mu Xutong, Cheng Ke, Song Anxiao, etc. Privacy preserving federated learning against Byzantine attacks. Journal of Computer Science, 2024, 47(4): 842-861 (in Chinese)
(穆旭彤，程珂，宋安霄，等.抗拜占庭攻击的隐私保护联邦学习.计算机学报，2024，47(4):842-861)

[8] Wang Ruijin, Wang Jinbo, Zhang Fengli, et al. The feature map poisoning attack and dual defense mechanism of federated prototype learning. Journal of Software, 2022, 32(01): 1-20 (in Chinese)
(王瑞锦，王金波，张凤荔，等.联邦原型学习的特征图中毒攻击和双重防御机制.软件学报，2022，32(01):1-20)

[9] Tang Lingtao, Chen Zuoning, Zhang Lufei, et al. Research progress on privacy issues in federated learning. Journal of Software, 2023, 34(01): 197-229 (in Chinese)
(汤凌韬，陈左宁，张鲁飞，等.联邦学习中的隐私问题研究进展.软件学报，2023，34(01):197-229)

[10] Zhang C, Yang S, Mao L, et al. Anomaly detection and defense techniques in federated learning: a comprehensive review. Artificial Intelligence Review, 2024, 57(6): 150-184

[11] Wang Yongkang, ZhaiDihua, Xia Yuanqing. Robust aggregation algorithm for resisting a large number of backdoor clients in federated learning. Journal of Computer Science, 2023, 46(6): 1302-131 (in Chinese)
(王永康，翟弟华，夏元清.联邦学习中抵抗大量后门客户端的鲁棒聚合算法.计算机学报，2023，46(6):1302-1314)

[12] Xie C, Chen M, Chen P Y, et al. Crfl: certifiably robust federated learning against backdoor attacks//Proceedings of the 2021International Conference on Machine Learning. Vienna, Austria, 2021: 11372-11382

[13] Zhang H, Li X, Xu M, et al. BADFL: backdoor attack defense in federated learning from local model perspective. IEEE Transactions on Knowledge and Data Engineering, 2024, 36(11): 5661-5674

[14] Gao Ying, Chen Xiaofeng, Zhang Yiyu, et al. A review of research on attack and defense techniques for federated learning systems. Journal of Computer Science, 2023, 46 (09): 1781-1805 (in Chinese)
(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.计算机学报,2023,46(09):1781-1805)

[15] Awan S, Luo B, Li F. Contra: defending against poisoning attacks in federated learning//Proceedings of the European Symposium on Research in Computer Security. Darmstadt, Germany, 2021: 455-475

[16] Xie C, Huang K, Chen P Y, et al. Dba: distributed backdoor attacks against federated learning//Proceedings of the International Conference on Learning Representations. Addis Ababa, Ethiopia, 2020: 1-19

[17] Li Z, Zhu Y, Van Leeuwen M. A survey on explainable anomaly detection. ACM Transactions on Knowledge Discovery from Data, 2023, 18(1): 1-54

[18] Nowroozi E, Haider I, Taheri R, et al. Federated learning under attack: exposing vulnerabilities through data poisoning attacks in computer networks. IEEE Transactions on Network and Service Management, 2025, 22(1): 822-831

[19] Gupta P, Yadav K, Gupta B B, et al. A novel data poisoning attack in federated learning based on inverted loss function. Computers & Security, 2023, 130: 103270

[20] Kasyap H, Tripathy S. Beyond data poisoning in federated learning. Expert Systems with Applications, 2024, 235: 121192

[21] Tolpegin V, Truex S, Gursoy M E, et al. Data poisoning attacks against federated learning systems//Proceedings of the 2020 European Symposium on Research in Computer Security. Guildford, UK, 2020: 480-501

[22] Yang H, Gu D, He J. A robust and efficient federated learning algorithm against adaptive model poisoning attacks. IEEE Internet of Things Journal, 2024, 11(9): 16289-16302

[23] Xiao Xiong, Tang Zhuo, Xiao Bin, etc A review of privacy protection and security defense research in federated learning. Journal of Computer Science, 2023, 46 (5): 1019-1044 (in Chinese)
(肖雄,唐卓,肖斌,等.联邦学习的隐私保护与安全防御研究综述.计算机学报,2023,46(5):1019-1044)

[24] Liu Jialang, Guo Yanming, Lao Mingrui, etc. Overview of backdoor attack and defense algorithms based on federated learning. Computer Research and Development, 2024, 61 (10): 2607-262 (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.计算机研究与发展,2024,61(10):2607-2626)

[25] Fang M, Cao X, Jia J, et al. Local model poisoning attacks to {Byzantine-Robust} federated learning//Proceedings of the USENIX Security Symposium. Boston, USA, 2020: 1605-1622

[26] Bagdasaryan E, Veit A, Hua Y, et al. How to backdoor federated learning//Proceedings of the International Conference on Artificial Intelligence and Statistics. Palermo, Italy, 2020: 2938-2948

[27] Cao X, Gong N Z. Mpaf: model poisoning attacks to federated learning based on fake clients//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, USA, 2022: 3396-3404

[28] Zhang J, Chen B, Cheng X, et al. PoisonGAN: generative poisoning attacks against federated learning in edge computing systems. IEEE Internet of Things Journal, 2020, 8(5): 3310-3322

[29] Blanchard P, El Mhamdi E M. Guerraoui R, et al. Machine learning with adversaries: Byzantine tolerant gradient descen//Proceedings of the Neural Information Processing Systems. Long Beach, USA, 2017: 30

[30] Xie C, Koyejo S, Gupta I. Zeno: distributed stochastic gradient descent with suspicion-based fault-tolerance//Proceedings of the International Conference on Machine Learning. Los Angeles, USA, 2019: 6893-6901

[31] Liu J, Li X, Liu X, et al. DefendFL: a privacy-preserving federated learning scheme against poisoning attacks. IEEE Transactions on Neural Networks and Learning Systems, 2024, 36(5), 9098-9111

[32] Lai Y C, Lin J Y, Lin Y D, et al. Two-phase defense against poisoning attacks on federated learning-based intrusion detection. Computers & Security, 2023, 129: 103205

[33] Andrew G, Thakkar O, McMahan B, et al. Differentially private learning with adaptive clipping//Proceedings of the Neural Information Processing Systems. Virtual, 2021, 34: 17455-17466

[34] Cao X, Fang M, Liu J, et al. FLTrust: Byzantine-robust federated learning via trust bootstrapping//Proceedings of the Network and Distributed System Security Symposium. Virtural, 2021: 1-18

[35] Guo Jingjing, Liu Jiuzun, Ma Yong, etc. A defense method for federated learning backdoor attacks based on model watermarking. Journal of Computer Science, 2024, 47 (3): 662-676 (in Chinese)
(郭晶晶,刘玖樽,马勇,等.基于模型水印的联邦学习后门攻击防御方法.计算机学报,2024,47(3):662-676)

[36] Purohit K, Das S, Bhattacharya S, et al. A data-driven defense against edge-case model poisoning attacks on federated learning//Proceedings of theEuropean Conference on Artificial Intelligence. Antiago de Compostela, Spain, 2024: 2162-2169

[37] Yan G, Wang H, Yuan X, et al. FedRoLA: robust federated learning against model poisoning via layer-based aggregation//Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining. Barcelona, Spain, 2024: 3667-3678

[38] Huang W, Ye M, Shi Z, et al. Parameter disparities dissection for backdoor defense in heterogeneous federated learning//Proceedings of the Neural Information Processing Systems. Vancouver, Canada, 2024, 37: 120951-120973

[39] Vilone G, Longo L. Notions of explainability and evaluation

approaches for explainable artificial intelligence. Information Fusion, 2021, 76: 89-106

[40] Černevičienė J, Kabašinskas A. Explainable artificial intelligence (XAI) in finance: a systematic literature review. Artificial Intelligence Review, 2024, 57(8): 216-261

[41] Dwivedi R, Dave D, Naik H, et al. Explainable AI (XAI): core ideas, techniques, and solutions. ACM Computing Surveys, 2023, 55(9): 1-33

[42] Abtahi A, Aminifar A. Privacy-preserving federated interpretability//Proceedings of the 2024 IEEE International Conference on Big Data. Shenzhen, China, 2024: 7592-7601

[43] Kalakoti R, Bahsi H, Nõmm S. Explainable federated learning for botnet detection in IoT networks//Proceedings of the 2024 IEEE International Conference on Cyber Security and Resilience.London, UK, 2024: 1-8

[44] Bárcena J L C, Ducange P, Marcelloni F, et al. Increasing trust in AI through privacy preservation and model explainability: federated learning of fuzzy regression trees. Information Fusion, 2025, 113: 102598

[45] Song T, Tong Y, Wei S. Profit allocation for federated learning// Proceedings of the 2019 IEEEInternationalConferenceon Big Data. Los Angeles, USA, 2019: 2577-2586

[46] Liu Z, Chen Y, Yu H, et al. Gtg-shapley: efficient and accurate participant contribution evaluation in federated learning. ACM Transactions on Intelligent Systems and Technology, 2022, 13(4): 1-21

[47] Sandeepa C, Siniarski B, Wang S, et al. SHERPA: explainable robust algorithms for privacy-preserved federated learning in future networks to defend against data poisoning attacks//Proceedings of the 2024 IEEE Symposium on Security and Privacy. San Francisco, USA, 2024: 204

[48] Khuu D P, Sober M, Kaaser D, et al. Data poisoning detection in federated learning//Proceedings of the 39th ACM/SIGAPP Symposium on Applied Computing. Avila, Spain, 2024: 1549-1558

[49] Sun Q, Zhang J, Liu J, et al. Shapley value approximation based on complementary contribution. IEEE Transactions on Knowledge and Data Engineering, 2024, 36(12), 9263-9281

[50] Kolpaczki P, Bengs V, Muschalik M, et al. Approximating the shapley value without marginal contributions//Proceedings of the AAAI Conference on Artificial Intelligence. Vancouver, Canada, 2024: 13246-13255

[51] Sant'Ana L, Cordeiro D, de Camargo R Y. PLB-HAC: dynamic load-balancing for heterogeneous accelerator cluster// Proceedings of the European Conference on Parallel Processing. Göttingen, Germany, 2019: 197-209

JIANG Wei-Jin, Ph. D., professor, Ph. D. supervisor. His research interests are edge computing, social computing, and, cyberspace security.

YANG Xuan, M. S. His research interests are federated learning and privacy protection.

LI Bi-Xia, M. S. Her research interests are federated learning and model security

Background

BackgroundThe research presented in this article belongs to the field of federated learning (FL), with a particular focus on addressing the challenges of defending against poisoning attacks in the presence of non independent and identically distributed data. Poisoning attacks pose a serious threat to federated learning systems, as malicious clients manipulate their local model updates to disrupt the performance of the global model. Due to the inherent heterogeneity of client data, this problem is further exacerbated in practical FL environments, which naturally leads to divergence in benign model updates. Distinguishing this benign divergence from malicious interference is a long-term challenge in this field.

At present, the international research community has made significant progress in mitigating poisoning attacks, mainly through anomaly detection, clustering based filtering, and similarity measurement between model updates. However, these techniques often rely on strong assumptions about data distribution and update consistency. In non IID settings, these assumptions will fail: benign updates may deviate significantly from each other, and when malicious clients make up a large portion of the participants, they may disrupt the statistical majority that many defenses rely on. In addition, current solutions typically apply a uniform pruning strategy to suspicious updates, which often leads to the removal of benign components mixed in with malicious updates. This indiscriminate pruning slows down convergence speed, reduces overall model performance, and limits the practical effectiveness of these defense measures.

To address these limitations, this paper proposes the Contribution Anomaly Detection and Clipping (CADC)

algorithm based on interpretable contributions. By introducing SHAP values to quantify the local contribution of individual model parameters, parameter level anomaly detection sensitive to subtle interference can be achieved. By analyzing the contribution distribution of each parameter, CADC effectively distinguishes between benign differences caused by data heterogeneity and intentional malicious interference. In addition, CADC adopts dynamic pruning algorithm to preserve the benign components in pollution updates and maintain the convergence and accuracy of the global model. This study is part of a broader research plan aimed at improving the security and credibility of federated learning frameworks.

This article has received funding from the National Natural Science Foundation of China (No. 61772196), the Key Scientific Research Projects of Hunan Provincial Department of Education (No. 24A0446, No. 24A0753), and the Graduate Research Innovation Project of Hunan University of Technology and Business (No. CX2024YB001).