小型微型计算机系统
Journal of Chinese Computer Systems
DOI:10.20009/j.cnki.21-1106/TP.2025-0114
2026年4月 第4期
Vol.47 No.4 2026

边缘联邦学习中的量化感知训练及安全挑战

雷程宇,吴黎兵,张壮壮,王恩澍,霍丽娟,冯佳琪

(武汉大学 国家网络安全学院,武汉 430000)
E-mail:wu@whu.edu.cn

摘 要: 传统量化模型虽能降低模型复杂度、降低推理开销,但其量化扰动也会带来一定的性能损失.量化感知训练旨在提高神经网络对量化扰动的鲁棒性.为了使联邦学习中的边缘设备在计算资源受限的情况下进行实时推理,本文将量化感知训练引入边缘联邦学习场景,并提出了量化感知边缘联邦学习框架.在该框架中,量化后的全局模型部署在终端,并且不会产生过大性能损失,从而解决了边缘设备对用户推理需求实时快速响应与自身算力不足的矛盾.此外,本文发现在联邦学习中引入量化感知训练会带来一定的安全风险,攻击者可以利用量化感知训练恶意模型.进一步地,本文也提出了两种联邦量化攻击.实验结果表明本文所提方法在CIFAR10数据集上,使用ResNet18训练的全局模型在被量化至4-bit时仍保持62%的准确率,相较于传统的边缘联邦学习方法提升30%.另外,联邦量化攻击在8-bit量化下的攻击成功率相比现有工作提升10%.

关 键 词: 联邦学习;量化扰动;边缘联邦学习;量化感知训练
中图分类号: TP389 **文献标识码:** A **文 章 编 号:** 1000-1220(2026)04-0927-10

Quantization-aware Training and Security Challenges in Edge Federated Learning

LEI Chengyu, WU Libing, ZHANG Zhuangzhuang, WANG Enshu, HUO Lijuan, FENG Jiaqi
(School of Cyber Science and Engineering, Wuhan University, Wuhan 430000, China)

Abstract: Traditional quantization models can reduce model complexity and inference costs, but their quantization perturbations can lead to performance losses. Quantization-aware training aims to improve neural networks' robustness to quantization perturbations. To enable edge devices in federated learning to perform real-time inference under limited computational resources, this study introduces quantization-aware training into the edge federated learning scenario and proposes a quantization-aware edge federated learning framework. In this framework, the quantized global model is deployed on the terminals without significant performance losses, addressing the contradiction between edge devices' real-time response to user inference demands and their insufficient computing power. Additionally, this study finds that introducing quantization-aware training into federated learning poses certain security risks, as attackers can exploit malicious models trained using quantization-aware training. Based on this, the study also proposes two federated quantization attacks. Experimental results demonstrate that the proposed approach maintains a 62% accuracy rate when a globally trained ResNet18 model is quantized to 4 bits on the CIFAR10 dataset, a 30% improvement over traditional edge federated learning methods. Furthermore, the success rate of federated quantization attacks increases by 10% under 8-bit quantization compared to existing works.

Keywords: federated learning; quantization perturbations; edge federated learning; quantization-aware training

0 引 言

神经网络的发展推动了人工智能在多个领域的重大突破.然而,随着模型参数规模的不断扩大,模型的复杂性也随之显著提升.随着参数数量的增加,对计算和存储资源的需求更高.为应对这些问题,模型压缩技术[1]已成为深度学习研究中的重要方向,其目标是在尽量保持模型精度的前提下,显著降低计算和存储成本.常见的模型压缩方法包括参数剪枝、量化和低秩分解等.其中,量化技术[2-4]被认为是最有效的手段之一.通过将模型参数从高精度浮点格式(如32-bit)转换为低位宽(low-bit)格式(如4-bit或更低),量化可以大幅减少模型的存储需求并降低推理过程中的计算复杂度.

联邦学习(Federated Learning,FL)作为一种分布式学习范式,因其在保护数据隐私和降低通信成本方面的优势而受到广泛关注.其中,FedAvg[5]是最经典的联邦学习算法,将全局模型分发给各客户端进行本地训练,客户端随后将更新上传至服务器,服务器再聚合所有客户端的模型更新生成新一轮的全局模型.尽管FedAvg为联邦学习的广泛应用奠定了基础,但其通信与计算成本始终是一个瓶颈,特别是在移动边缘网络(Mobile Edge Networks,MEC)场景下,终端设备通常

收稿日期:2025-03-10 收修改稿日期:2025-04-09 基金项目:国家重点研发计划项目(2022YFB3104502)资助;武汉市星地融合新一代无线通信产业创新联合实验室项目(4050902040448)资助;武汉市科技计划项目(2024050702030090)资助;武汉市交通强国建设试点科技联合项目(No.2023-2-7)资助. 作者简介:**雷程宇**,男,1999年生,硕士研究生,研究方向为数据安全、联邦学习;**吴黎兵**(通信作者),男,1972年生,教授,博士生导师,CCF杰出会员,研究方向为物联网、网络安全、数据安全等;**张壮壮**,男,1994年生,博士研究生,CCF会员,研究方向为机器学习安全、数据安全;**王恩澍**,男,1990年生,博士,教授,CCF会员,研究方向为强化学习、车联网安全;**霍丽娟**,女,1997年生,博士研究生,CCF学生会员,研究方向为数据安全、网络安全;**冯佳琪**,女,2000年生,博士研究生,研究方向为数据安全、网络安全.

具有有限的带宽和算力.

为降低联邦学习中的通信开销,不少研究将量化技术引入其中. 例如,FedPAQ[6]在客户端上传本地模型更新之前对其进行量化处理,以减少传输的数据量. Tonellotto 等人提出的 FLQ 算法[7]则进一步对客户端的模型更新和服务端的全局模型进行双向量化,从而优化了整体通信效率. 然而,现有将联邦学习与量化相结合的研究主要集中在训练阶段的通信成本优化,较少关注联邦学习模型部署后的量化问题. 事实上,联邦学习模型部署阶段的量化问题同样重要:联邦学习的一个重要应用领域是基于物联网的终端设备(例如手机、智能手表、家用传感器等)[8]. 这种场景又称为边缘联邦学习,对于边缘设备而言,训练阶段的工作可以后台异步进行,而部署阶段的推理往往需要实时响应用户输入,如输入法的文本预测[9]、图片识别[10]等任务. 用户对推理时延的敏感性使得降低推理延迟成为关键问题. 物联网终端设备因为计算和存储资源受限,在使用复杂的神经网络模型推理时无法保障实时性.

基于这一难题,本文对部署后的联邦学习全局模型进行量化以减少终端设备在推理时的计算开销. 然而,传统的后量化方法(Post-Training Quantization, PTQ)在较低位宽(如 4-bit)下通常会导致显著的性能下降. 为解决这一问题,量化感知训练(Quantization-Aware Training, QAT)被提出并广泛应用. QAT 在模型训练阶段模拟实际量化环境,使模型在未来可能的低精度推理条件下学习更具鲁棒性的参数分布[6]. 这一方法显著缓解了量化带来的精度损失问题,并在低位宽场景下表现出卓越性能. 例如,Jacob 等人[11]在 ImageNet 数据集上使用 QAT 对 ResNet-50 进行优化,在 8-bit 量化条件下实现了接近全精度模型的性能. 本文在联邦学习的训练阶段引入 QAT 来降低全局模型在部署时因量化而带来的性能损失. 本文提出了量化感知联邦学习(QAFed)框架,在本文的框架中,服务端和各边缘服务器首先需要协商目标量化比特来选举出最符合终端设备需求的量化方式,然后在训练阶段依据提前协商好的目标量化比特来进行量化感知训练,在损失函数中引入量化函数以学习对量化具有鲁棒性的参数分布. 部署阶段各客户端依据目标量化比特对自己得到的最终模型进行量化.

此外,虽然在联邦学习中较少有研究关注与量化部署和量化感知训练(QAT)相关的问题,但是利用量化感知训练来实施非分布式场景下的传统模型后门攻击的案例却已经被提出,例如,量化后门攻击[12]通过优化后门模型,使其在量化后的低位宽条件下依然有效,同时保持触发条件的隐蔽性. 量化模型退化攻击则利用量化过程中的参数敏感性,通过精心设计的扰动削弱模型的整体性能. 因此,本文也研究了在联邦学习中引入 QAT 所带来的安全隐患.

本文贡献如下:

1)提出量化感知边缘联邦学习框架并验证其实用性. 针对物联网终端设备资源受限但高度依赖实时推理的需求,本文首次提出了量化感知边缘联邦学习框架. 在该框架中本文定义了多比特量化感知联邦学习的优化目标和损失函数. 服务端与边缘服务器协商获得最符合实际需求的量化比特集合. 对于攻击者,本文提出了一个目标量化比特预测算法,通过预测全局模型的目标比特集合,有针对性地引导攻击模型的优化方向. 实验结果表明该框架显著提升了模型对量化扰动的鲁棒性,并支持在带宽和计算资源受限的环境中高效部署.

2)定义并研究量化感知联邦学习中的两类攻击. 本文分析了量化感知边缘联邦学习框架下的威胁模型. 将量化感知训练武器化,在框架中引入恶意客户端,并定义了量化模型退化攻击和量化后门攻击两种攻击方式. 实验结果表明,在最终模型部署时,攻击模型在量化之前不表现明显的恶意行为,而量化之后攻击效果显著激活,展现了极高的隐蔽性.

3)评估现有防御方法并探讨对策. 本文评估了两种常见的联邦学习后门防御方法,并通过对抗性训练增强恶意客户端对防御机制的对抗能力. 基于实验结果,结合传统神经网络后门检测工作,本文提出了针对服务端安全聚合的建议,以提升联邦学习系统的整体安全性.

1 相关工作

本文的研究基于三方面的工作:量化感知训练、联邦学习中的量化和后门攻击.

1.1 量化感知训练

量化感知训练(Quantization-Aware Training, QAT)是一种有监督的训练方法,通过在训练时量化神经网络中的参数和激活值来模拟量化环境. 标准的量化方法是 STE(直通估计器)[13]. 需要注意的是,使用这种方法进行量化感知训练时,在反向传播时需要对量化过程中的不可微操作进行梯度估计造成梯度不匹配问题,这可能会导致优化过程的不稳定. 针对此问题,一些工作设计了更先进的可微量化方法[14-16],来改进量化感知训练的优化. 也有一些方法以最小量化误差为目标,这会减少因量化扰动而对模型带来的性能损失[17-19]. 本文采用最基础的量化方法 STE[14]以在训练阶段模拟对模型影响最大的量化扰动,从而尽可能获得更大的鲁棒性.

1.2 联邦学习中的量化

在联邦学习的框架中,大量研究通过量化技术来减少训练过程中客户端与服务器之间的通信量[6,7,20,21],这样做能够加快训练速度,节省训练成本. 但上传模型更新之前的量化操作也会引入来自各个客户端模型更新的累计量化误差,从而导致训练模型性能下降. 本文致力于研究联邦模型在训练完成后设备端的推理成本和表现,并且本文使用的方法(量化感知训练)在训练阶段只是在损失函数中模拟量化扰动,并不需要真的量化模型更新,从而并不导致性能损失.

1.3 后门攻击

在本文场景的威胁模型中,与后门攻击相关的以往工作主要分为两类. 一方面是针对量化模型的后门攻击[12,22]. 此类研究的核心目标是通过训练一个后门模型,使其在量化前不表现出明显的恶意行为,而在量化后通过触发机制激活恶意特征,从而实施攻击. 这类工作通常聚焦于模型供应链安全和模型外包等集中式场景,未充分考虑分布式环境中的复杂性和安全隐患.

另一方面,联邦学习由于采用安全聚合机制(Secure Aggregation)[23],在保护客户端隐私的同时,也阻止了服务器对

客户端模型更新的直接审查,为恶意客户端实施后门攻击提供了便利. 因此,联邦学习环境下的后门攻击成为近年来的研究热点. 例如,Bagdasaryan 等人[24]提出了通过精心设计的本地更新,在全局模型中隐蔽植入后门的攻击方法;Fung 等人[25]研究了通过 Sybil 攻击加强后门效果的场景;Xie 等人[26]则提出了动态模型替换攻击(Dynamic Backdoor Attack),展示了更强的攻击灵活性和隐蔽性. 这些研究表明,联邦学习框架中的分布式特性进一步增加了检测和防御后门攻击的难度,但现有研究大多集中在全精度模型,尚未探索量化感知训练对联邦后门攻击的潜在影响.

表 1　联邦学习中的后门攻击研究概览

Table 1　Survey of backdoor attacks in federated learning

研究名称	是否考虑分布式场景	是否利用量化进行攻击
PQ Backdoor[12]	×	√
Qu-anti-zation[22]	×	√
Model Replacement[24]	√	×
FoolsGold[25]	√	×
DBA[26]	√	×
本文工作	√	√

表 1 总结了上述工作,相比之下,本文提出的攻击同时考虑了分布式场景和利用量化漏洞进行攻击,填补了相关空白.

2　算法框架

在本节中,本文介绍了量化感知边缘联邦学习的框架,并详细阐述了如何将量化感知训练引入神经网络和边缘联邦学习中. 首先,本文讨论了量化鲁棒性神经网络的优化目标和量化过程,提出了通过直通估计器实现的量化函数. 接着,提出了一个多比特量化鲁棒性神经网络框架,旨在优化模型在不同量化比特下的表现,以满足不同硬件环境下的部署需求. 本文进一步将量化感知训练引入边缘联邦学习场景,设计了一个基于 QAT 的联邦学习框架(QAFed),如图 1 所示,通过多比特量化策略和量化感知训练,提升了模型对量化扰动的鲁棒性. 最后,本文讨论了所提出框架下的威胁模型,详细分析了攻击者和防御者的能力,并提出了相应的防御策略.

2.1　量化鲁棒性神经网络

假设有神经网络模型 f,模型参数为 w,目标量化比特为 b,数据集是 D,F 表示损失函数,采用交叉熵损失. 传统神经网络的优化目标损失为:

$$\min F(w,D) \tag{1}$$

其中:

$$F(w,D)=\mathbb{E}_{(x,y)\sim D}[\mathcal{L}_{ce}(f(x;w),y)] \tag{2}$$

量化鲁棒性神经网络算法旨在训练一个模型,其对不同比特的量化扰动具备鲁棒性,在量化之前和量化之后的性能不会有明显的区别. 则量化鲁棒性神经网络的目标是同时优化以下损失函数:

$$\min F(Q_b(w),D),F(w,D) \tag{3}$$

其中:

$$F(Q_b(w),D)=\mathbb{E}_{(x,y)\sim D}[\mathcal{L}_{ce}(f(x;Q_b(w)),y)] \tag{4}$$

本文采用直通估计器作为量化函数,即:

$$Q_b(w)=\sum_{l=1}^{L}\frac{\mathrm{clip}(\lfloor w_l\cdot s_b^l\rfloor,-2^{b-1},2^{b-1}-1)}{s_b^l} \tag{5}$$

其中$\lfloor\cdot\rfloor$表示四舍五入操作,clip(·)表示将输入限制在$[-2^{b-1},2^{b-1}-1]$,w_l 表示第 1 层参数,s_b^l 是 b 比特量化下第 l 层参数的缩放因子. 本文根据每一层的神经网络参数动态地确定缩放因子,即采用逐层量化的方式:

$$s_b^l=\frac{(2^{b-1}-1)-(-2^{b-1})}{\max(w_l)-\min(w_l)} \tag{6}$$

公式(6)中的缩放因子可以将原参数归一化,然后再放大到目标比特的范围. 再通过公式(5)中的四舍五入操作用整数表示参数. 最终除以缩放因子是为了在训练阶段方便现有神经网络框架的浮点运算[12],但总体参数分布模拟的仍是推理阶段量化到目标比特后的参数分布. 考虑到不同终端设备的部署环境不同,量化目标比特也有可能不同,为了部署时在不同量化比特下都具备良好的性能表现,最终本文使用多比特量化鲁棒性神经网络,其优化目标损失如下:

$$\min\sum_b F(Q_b(w),D) \tag{7}$$

特别的,当 $b=32$ 时表示的就是公式(1)中的优化任务. 为此,本文使用量化感知训练来达成最小化目标损失,即在正向传播时,使用公式(5)量化神经网络参数,计算出目标损失(7),反向传播时则关闭量化模块,并使用全精度参数来优化目标损失.

2.2　量化感知边缘联邦学习

在边缘联邦学习场景中,由于终端设备通常受限于计算和存储资源,而用户对推理阶段的实时性需求显著高于对训练阶段的速度要求,因此本文更加关注联邦模型在部署到终端设备后其推理性能的表现. 为此,本文提出了一种基于量化感知训练的边缘联邦学习框架,用于提升部署模型在量化环境下的鲁棒性和性能.

如图 1 所示,在联邦学习的初始化阶段,边缘服务器首先对终端设备的硬件算力条件(例如计算能力、内存带宽等)进行评估,以确定适合的量化比特范围. 随后,中央服务器与边缘服务器通过加权投票机制协商生成目标量化比特集合 B. 具体流程如下:1)终端设备评估:每个边缘服务器收集管理设备的计算能力、存储带宽、数据质量等信息,并生成推荐的量化比特集合 B;2)边缘服务器投票:各边缘服务器提交自己的量化比特建议. 投票权重由服务器的设备数量、计算能力、数据质量等因素决定;3)中央服务器决策:选出得票最高的前两种量化比特方案作为最终方案;4)比特集合下发:中央服务器向边缘服务器下发 B,边缘服务器再分发至终端设备,指导后续训练. 在此过程中系统采用身份认证与加密通信协议防范潜在攻击者通过窃取或篡改量化比特信息设计针对性的量化后门攻击. 选举过程完成后,中央服务器将目标量化比特集合 B 下发至参与的终端设备,并正式启动联邦量化感知训练(Quantization-Aware Federated Learning,QAFed)

训练阶段,本文将式(7)中的目标损失迁移到联邦学习的场景中,得到如下全局目标损失:

$$\min_{\mathbf{w}}\mathbb{E}_{i\sim P}[F_i(\mathbf{w}_i,D_i)+\sum_{b\in B}F_i(Q_b(\mathbf{w}_i),D_i)] \tag{8}$$

其中:

$$F_i(Q_b(\mathbf{w}_i),D_i)=\mathbb{E}_{(x,y)\sim D_i}[\mathcal{L}_{ce}(f_i(x;Q_b(w_i)),y)] \tag{9}$$

依据损失函数(8),本文提出了 FedAvg[5] 算法的变体:

量化感知联邦训练(Quantization-Aware Federated Learning, QAFed). QAFed 的伪代码如算法 1 所示. 初始全局模型表示为 $\mathbf{w}_0$,全局学习率用 η_s 表示,本地学习率 η_c,全局训练轮次 T,本地训练轮次 E,客户端总数(非选中数量)N. 该算法运行于边缘联邦学习架构中,结合量化感知训练机制,旨在提升全局模型对量化扰动的鲁棒性,同时优化分布式训练的效率.

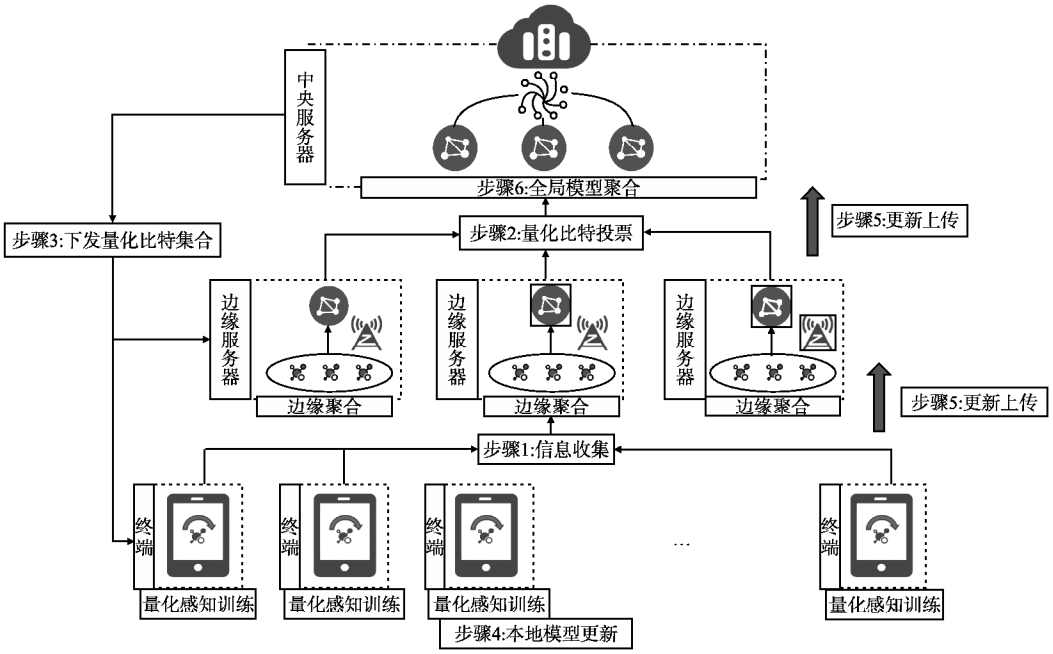

图 1 边缘联邦学习下的量化感知训练

Fig. 1 Quantization-aware training in edge federated learning

在每轮联邦学习开始之前,中央服务器对边缘服务器和终端设备进行动态选择:中央服务器根据各边缘服务器的计算能力、带宽资源和上一轮的模型贡献程度,按照一定比例筛选出参与本轮联邦学习的边缘服务器. 随后,每个被选中的边缘服务器进一步筛选其管理下的终端设备,依据设备的硬件能力(如算力、内存带宽)和数据质量(如样本量和多样性),按照比例确定参与本轮训练的终端设备.

算法 1. Quantization-Aware Federated Learning (QAFed)

Input: $\mathbf{w}_0, B, \eta_c, \eta_s, T, E$.

Output: Optimized global model $\mathbf{w}_T$

FOR each round $t = 1, 2, \cdots, T$ **DO**
  Central Server: sample a subset S_t of edge servers.
  FOR $s \in S_t$ **DO**
    Edge Server s: sample a subset C_s of clients.
    FOR each selected client $i \in C_s$ **DO**
      $\mathbf{w}_{t,0}^i \leftarrow \mathbf{w}_t$
      FOR local epoch $e = 1, 2, \cdots, E$ **DO**
        Enable Quantize Module:
        Initialize loss $L \leftarrow 0$
        FOR each $b \in B$ **DO**
          $L \leftarrow \mathcal{L}_{ce}(f_i(x_i; Q_b(\mathbf{w}_{t,e}^i)), y_i)$
        ENDFOR
        Disable Quantize Module:
        $g_{t,e}^i \leftarrow \nabla_{\mathbf{w}} L$
        $\mathbf{w}_{t,e+1}^i \leftarrow \mathbf{w}_{t,e}^i - \eta_c \cdot g_{t,e}^i$
      ENDFOR
    ENDFOR
  ENDFOR
  $\Delta_s = \sum_{i \in Cs} (\mathbf{w}_{t,E}^i - \mathbf{w}_{t,0}^i)$
ENDFOR
$\Delta_t = \sum_{s \in S_t} \Delta_s$
$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + \frac{\eta_s \cdot \Delta_t}{N}$
ENDFOR

在 QAFed 的正向传播阶段,各终端设备基于筛选结果参与训练,并按照协商的目标量化比特集合 B 对神经网络参数进行量化,模拟推理环境中的量化噪声,计算损失(9). 反向传播阶段则保持参数的全精度表示,通过在神经网络的各层嵌入量化模块实现动态控制:量化模块在正向传播时激活,用于处理权重和激活值,而在反向传播时禁用,使梯度计算基于全精度参数表示. 这一机制使模型能够在训练阶段不断优化损失函数(8),学习到更具鲁棒性的参数分布,从而显著降低因低精度量化引起的性能退化问题.

训练完成后,各终端设备将本地计算的梯度上传至其对应的边缘服务器. 边缘服务器对接收到的终端设备模型更新进行局部聚合,以减少中央服务器的通信负担. 随后,边缘服务器将聚合后的模型更新上传至中央服务器,中央服务器对所有边缘服务器的更新进行全局聚合,并生成优化后的全局模型. 完成聚合后,全局模型被下发至边缘服务器和终端设备,为下一轮训练迭代提供优化初始值. 最终,经过量化感知训练的全局模型在终端设备上进一步量化压缩,以满足实时推理的性能需求,同时保证在资源受限环境下的高效部署. 本文的框架通过动态量化策略和量化感知训练,能够在资源受限的 IoT 设备上实现高效的模型部署,对比传统联邦学习算

法[5],本文显著提升模型在量化环境中的推理性能. 具体实验数据见 4.2 小节.

2.3 威胁模型

由于联邦学习的去中心化特点,模型训练过程分布在多个终端设备上,攻击者可以通过恶意客户端插入后门或干扰模型性能,进而对全局模型的行为产生严重影响. 在资源受限的边缘计算场景中,模型量化是提升推理效率和降低存储需求的重要技术. 然而,量化技术也为攻击者提供了新的攻击面. 攻击者可以利用量化过程中模型参数的分布变化,设计出在量化前不表现恶意行为但量化后恶意行为被激活的攻击策略. 本文在第 4 节具体描述本文的攻击.

2.3.1 攻击种类

本文模拟实现了两种联邦量化攻击:联邦量化后门攻击(Federated Quantified Backdoor,FQBD)、联邦量化模型退化攻击(Federation Quantization Model Aegradation Attack,FQMA). 他们分别可以使量化后的模型对带有后门标签的样本分类到目标类别,以及对所有样本的准确率下降. 其中 FQBD 的关键指标是攻击成功率(Attack Success Rate,ASR),即将带后门特征的样本分类为目标类别的比率和干净数据准确率(Clean Data Accuracy,CDA)即对所有正常样本分类正确的比率. 攻击者的目标是得到一个休眠的全精度后门模型,具有高 ASR 和 CDA,该特征在量化后被激活. 而 FQMA 的关键指标是干净数据准确率(Clean Data Accuracy,CDA),攻击者的目标是导致目标模型显著的准确率下降,即具有低 CDA,该特征在量化后被激活.

2.3.2 攻击者的能力

- 恶意客户端的控制权. 攻击者可以控制联邦学习中的若干客户端,操控其训练数据和训练过程. 恶意客户端能够注入后门样本、干扰模型参数更新,并通过本地训练阶段构造在量化后表现出恶意行为的模型. 此外,攻击者可以在本地模拟模型量化过程,以测试并优化其攻击策略,使攻击效果最大化.
- 对量化比特的有限知情. 攻击者能够部分获知联邦学习中目标量化比特集合 的分布情况. 例如,通过推测终端设备的硬件性能,攻击者可能得知某些目标量化比特范围,但不能完全掌握全部细节. 这种有限的知情能力使攻击者能够更精准地设计适用于目标环境的量化后门或退化攻击.
- 受限的全局模型视图. 攻击者仅能通过参与联邦学习的本地模型更新过程间接推测全局模型的状态,而无法直接访问或修改全局模型. 尽管如此,攻击者可以利用联邦学习的迭代特性,通过多轮更新逐步对全局模型施加影响.
- 不可检测的行为隐蔽性. 攻击者设计的模型在量化前表现正常,与普通客户端无异,使其难以被服务器检测到异常行为. 这种能力依赖于攻击者对量化过程的理解,以及对联邦学习机制的适配. 攻击者的目标是在量化后激活恶意行为,同时规避现有的异常检测和防御策略.

2.3.3 防御者的能力

在量化感知边缘联邦学习的场景中,防御者需要应对量化后门攻击和模型退化攻击的潜在威胁,具备以下两方面的能力以保障全局模型的安全性和鲁棒性:

- 从联邦学习的角度对模型更新进行处理. 防御者可以利用联邦学习中的分布式特性,对客户端上传的模型更新进行筛选和聚合. 例如,通过异常检测算法识别恶意客户端的更新,排除可能对全局模型产生负面影响的更新[27]. 此外,防御者可以通过调整聚合规则来增强全局模型对恶意更新的鲁棒性. 例如,基于更新值的大小进行裁剪,或对模型梯度进行噪声添加以掩盖恶意更新(Differential Privacy)[28].
- 从传统后门检测的角度对训练完成的模型进行后门检测. 防御者可以在全局模型收敛后,对其进行全面的后门检测,以评估模型中是否隐藏恶意行为. 例如,基于异常激活值的检测 MNTD[29],ABS[30] 通过测试特定触发模式下的模型输出是否异常,从而识别潜在的后门;基于触发器检测 Neural Cleanse[31],STRIP[32] 通过检测后门触发器对原始样本带来的扰动大小来判断模型的输出是否异常.

3 联邦量化攻击

3.1 量化感知武器化

Hong 等人[22]在传统模型供应链场景中将量化感知训练应用于后门攻击,成功实现了后门效果在量化后的激活,而在量化前保持潜伏状态. 借鉴这一思路,本文将其扩展至联邦学习的分布式环境,并通过多比特量化感知训练(Multi-Bit Quantization-Aware Training)提高攻击模型对不同量化精度的鲁棒性. 针对联邦量化模型退化攻击(Federated Quantization Model Attack,FQMA),本文设计了如下目标损失:

$$\min_{\mathbf{w}} \mathbb{E}_{i\sim A}\left[F_i(\mathbf{w}_i,D_i)+\lambda\cdot\left(\alpha-\sum_{b\in B'}F_i(Q_b(\mathbf{w}_i),D_i)\right)\right] \tag{10}$$

公式(10)中,第 1 项用于保证模型在量化前表现出正常行为,而第 2 项通过对量化后正常样本分类任务的惩罚,导致量化后模型的分类准确率 CDA 显著下降,从而实现攻击目标. 其中,λ 是用于平衡两部分损失的权重因子,α 确定惩罚项的上限,使对量化后分类性能的惩罚接近 α. A 表示全体恶意客户端.

由于恶意客户端无法完全掌握服务端的模型聚合过程,因此目标量化比特集合 B 的具体分布难以直接获取. B' 表示恶意客户端对 B 的预测结果. 预测的准确度决定了恶意模型在部署后应对量化扰动的鲁棒性. 预测越精确,模型在实际量化部署场景中的攻击效果越显著. 下一小节将具体描述此问题.

对于联邦量化后门攻击 FQBD,优化目标如下:

$$\min_{\mathbf{w}} \mathbb{E}_{i\sim A}\left[F_i(\mathbf{w}_i,D_i)+\lambda\cdot\sum_{b\in B'}F_i(Q_b(\mathbf{w}_i,D_i^{clean}))+F_i(Q_b(\mathbf{w}_i,D_i^{backdoor}))\right] \tag{11}$$

其中 λ 为调节因子,D_i^{clean} 表示干净数据集,$D_i^{backdoor}$ 表示后门数据集. 量化前的部分保持不变,攻击者将数据集分为两部分,一部分带有后门标签,一部分为正常干净数据集. 在优化目标损失中对干净数据分类任务和后门任务同时进行奖励,以达到量化之前保持隐蔽,量化之后对后门标签敏感的效果,具有高攻击成功率 ASR 和高分类准确率 CDA. 相比模型退化攻击,后门攻击更具有隐蔽性.

3.2 边缘联邦学习下的量化攻击

在边缘联邦学习框架下,由于终端设备的计算和存储资

源受限,量化技术成为模型部署的重要手段.然而,这也为攻击者提供了新的攻击面.在本节中,本文探讨联邦量化后门攻击(FQBD)和联邦量化模型退化攻击(FQMA)在边缘联邦学习环境中的具体实施步骤.

在联邦学习的初始阶段,服务器与各客户端协商目标量化比特集合 B(如 8-bit、6-bit、4-bit),以确保全局模型能够适应终端设备的算力条件.然而,由于服务器不会直接向客户端暴露具体的量化策略,恶意客户端无法直接获取 B 的完整信息.因此,为了提高攻击的有效性,恶意客户端采取延迟攻击策略(Delayed Attack Strategy),即在前若干轮全局训练迭代中保持静默,并通过主动测试全局模型的行为来推测目标量化比特集合 B,这种方式也可以避免攻击者的更新在前期被其他客户端的更新稀释,因为在前期各个客户端产生的模型更新幅度较大.攻击流程可分为以下 3 个阶段:

1)静默期(Silent Phase)

在联邦学习的初始轮次,恶意客户端保持静默,不发送恶意更新.同时,计算全局模型在每个比特下的平均准确率衰减,便于在启动攻击时预测目标比特集合:

在每一轮全局模型更新后,测试不同量化比特 $b \in B$ 下的模型准确率:

$$\mathrm{Acc}_b^t = \mathrm{Acc}(Q_b(\mathbf{w}_t), D_i^{clean}) \tag{12}$$

计算每个比特 b 的准确率衰减:

$$\Delta\ \mathrm{Acc}_b^t = \mathrm{Acc}_b^t - \mathrm{Acc}(\mathbf{w}_t, D_i^{clean}) \tag{13}$$

计算服务器可能选择的平均准确率衰减水平:

$$\mu_B = \mathbb{E}_t[\sum_{b\in B} \Delta\ \mathrm{Acc}_b^t] \tag{14}$$

2)目标比特预测(Target Bit Prediction)

恶意客户端基于静默期计算的平均准确率衰减,预测服务器最终可能采用的目标比特集合 B':

$$B' = \mathrm{argmax}_{B_{\text{test}}} \sum_{b\in B_{\text{test}}} -(\Delta\ \mathrm{Acc}_b^t - \mu_B) \tag{15}$$

其中:B_{test}是攻击者测试的备选量化比特集合,$\Delta\ \mathrm{Acc}_b^t$ 是比特 b 在当前轮次 t 造成的准确率衰减,μ_B 是服务器可能选择的稳定衰减水平,攻击者希望预测到最可能的量化比特集合.

3)启动攻击(Attack Execution)

当目标比特集合 B'确定后,恶意客户端开始发送恶意更新.对于联邦量化模型退化攻击(FQMA)采用公式(10)所示的目标损失来优化恶意模型.对于联邦量化后门攻击(FQBD)采用(11)来优化恶意模型.

$$\mathbf{w}_{i,e+1}^i \leftarrow \mathbf{w}_{i,e}^i + \eta_a \cdot \gamma \cdot g_{i,e}^i \tag{16}$$

其中 γ 是攻击放大因子,$g_{i,e}^i$是第 t 轮全局迭代,e 轮本地迭代模型的平均梯度更新.恶意客户端用公式(16)代替算法 1 中第 15 行的本地模型更新,其中 η_a 是攻击者本地学习率.为了提升攻击效果,恶意更新按照的方法进行放大:

$$\gamma = \frac{N}{\eta_s} \tag{17}$$

3.3 防御规避与对抗训练

为了提高攻击模型的隐蔽性并对抗这些防御措施,本文在攻击损失函数中引入高斯噪声扰动(Gaussian Noise Perturbation)进行对抗训练,并施加梯度更新约束(Gradient Constraint)以限制恶意梯度的幅度,使其更难以被异常检测识别.

1)对抗性训练目标

为增强攻击模型在防御机制下的有效性,本文修改攻击优化目标,对模型权重注入高斯噪声,以模拟差分隐私对梯度更新的影响.优化目标如下:

$$\min_{\mathbf{w}} \mathbb{E}_{i\sim A}[\boldsymbol{F}_i(\mathbf{w}_i', D_i) + \lambda \cdot \sum_{b\in B'} F_i(Q_b(\mathbf{w}_i', D_i^{clean})) + F_i(Q_b(\mathbf{w}_i', D_i^{backdoor}))] \tag{18}$$

其中:

$$\mathbf{w}_i' = \mathbf{w}_i + \sigma \cdot \mathrm{N}(0, I) \tag{19}$$

$\mathrm{N}(0,I)$代表均值为 0、协方差矩阵为单位矩阵的高斯噪声,σ 控制噪声强度.该噪声仅在正向传播时应用于模型权重$\mathbf{w}_i$,使得攻击模型在不同量化比特下能够适应差分隐私扰动,从而提升攻击成功率(ASR).在反向传播过程中,仍然使用原始权重$\mathbf{w}_i$ 计算梯度,以保证优化过程的稳定性和收敛性.

2)梯度更新约束

在每轮攻击训练中,本文对恶意梯度施加约束,确保其更新幅度不会超出设定的上限,以防止梯度异常导致攻击行为被检测到.具体而言,本文采用最大梯度范数约束(Gradient Norm Clipping):

$$g_i \leftarrow g_i \cdot \min\left(1, \frac{\tau}{\| g_i \|_2}\right) \tag{20}$$

其中,g_i 表示客户端 i 的局部梯度,τ 是设定的最大梯度范数.此操作确保攻击梯度不会过大,从而降低被异常检测算法检测到的概率.

3)防御机制评估

本文使用以下两种典型防御方法进行评测:全局梯度裁剪(Global Gradient Clipping):服务器在每轮联邦学习聚合梯度更新时,计算所有客户端梯度的 L_2 范数,并裁剪超过设定阈值的更新.差分隐私(Differential Privacy,DP):服务器在聚合前向客户端更新中添加高斯噪声,以保护数据隐私并抑制异常更新.

针对不同的防御强度,本文在 4.2.3 节进行了详细的实验分析,以验证攻击在防御机制下的成功率(Attack Success Rate,ASR)以及分类准确率(Clean Data Accuracy,CDA).

4 实验评估

通过标准的联邦学习数据集,本文进行了 3 种实验.实验 1 中,本文评估了量化感知边缘联邦学习框架在不同量化比特下的有效性,选取基本的联邦学习算法[5]作为 BaseLinc.实验 2 中,本文测试了本文提出的两种量化攻击在本文所提出框架下的表现,Hong 等人[22]将量化感知训练武器化训练后门并迁移到联邦学习场景,是当前最新的联邦量化攻击的研究,本文以他们的实验方法作为对比方案.实验 3 中,本文模拟了服务端采取梯度裁剪和差分隐私防御方法下量化后门攻击的对抗性表现.最后,本文结合以往工作,给出了对此类攻击的防御建议.

4.1 实验设置

量化感知联邦学习框架及其攻击使用 Pytorch 2.5.1 实现并进行测试,Python 版本是 3.12.本文使用两种数据集:CIFAR10 和 TinyImageNet.模型分别是 AlexNet 和 ResNet18.由于不同相近目标量化比特下模型的拟合曲线高度相似,为便于查看,本文主要使用三线表给出实验数据.对于分类准确率

(Clean Data Accuracy, CDA)和攻击成功率(Attack Success Rate, ASR),本文在表中分别给出测试周期内最高的数值. 硬件平台环境如下:CPU:Intel(R) Xeon(R) Platinum 8352V CPU @ 2.10GHz;GPU:NVIDIA RTX GeForce 4090(24GB) RAM:80GB.

4.2 实验结果

4.2.1 量化感知联邦学习

基本联邦学习算法[5]使用如下优化目标:

$$\min_{\mathbf{w}} \mathrm{E}_{i\sim P}[F_i(\mathbf{w}_i, D_i)] \tag{21}$$

本文的方案则使用公式(8)作为目标损失来优化全局模型,按照算法 1 中的步骤进行联邦模型训练. 总共有 100 个参与客户端,服务器每轮全局迭代随机挑选 10 个客户端参加模型训练,约定目标量化比特集合为 ,全局模型共进行 3000 次迭代,不同数据集和模型对应的超参数设置见表 2. 本文的实验结果见表 3.

表 2　超参数设置

Table 2　Hyperparameter settings

	η_s	η_c	E	BS
AlexNet + CIFAR10	9	0.008	5	128
AlexNet + TinyImageNet	4	0.003	7	128
ResNet18 + CIFAR10	6	0.008	5	128
ResNet18 + TinyImageNet	3	0.004	6	128

注:在本文的算法 QAFed 中,服务端聚合时,对模型更新乘以表中 η_s 服务器的学习率后还要除以客户端总数 N,BS 表示本地批量大小

表 3　量化感知联邦学习分类准确率(CDA%)

Table 3　Quantization-aware federated learning classification accuracy (CDA%)

Dataset	Bits	ResNet18		AlexNet	
		QAFed	BL	QAFed	BL
CIFAR10	32-bit	83.80%	86.20%	81.70%	84.30%
	8-bit	80.70%	66.10%	81.50%	67.50%
	4-bit	62.70%	48.30%	32.40%	30.10%
Tiny ImageNet	32-bit	60.30%	69.90%	60.30%	62.50%
	8-bit	60.20%	49.10%	60.20%	59.10%
	4-bit	49.50%	37.10%	39.50%	25.90%

注:QAFed 表示量化感知联邦学习算法,BL 表示 BaseLine 基本联邦学习算法

实验结果显示,本文的模型在量化到 8 比特时性能几乎没有损失,而量化到 4 比特最大 CDA 损失为 36%,对比 Baseline 算法,在量化到 8 比特时最大 CDA 损失 20%,量化到 4 比特时最大 CDA 损失 49.4%,本文的模型在面对量化扰动时展现了更高鲁棒性和稳定性. 图 2 展示了本文的模型和 Baseline 算法在 8 比特下的模型优化曲线,结果显示本文的模型拟合速度与 Baseline 模型基本相同,量化感知训练没有对模型的优化过程带来不稳定的影响.

4.2.2 联邦量化攻击

在本实验中,本文将 Hong 等人[22]的攻击模型作为 BaseLine,相比他们的模型,本文使用了多比特量化攻击,并主动预测服务端的目标量化比特,此外,本文在保障不产生明显异常的情况下放大了攻击者的恶意更新. 本文和 BaseLine 采用相同的攻击设定,在实验一原有联邦设置的基础上,假设攻击者控制 100 个客户端其中 5 个,服务器仍然每轮随机挑选 10 个客户端参加训练.

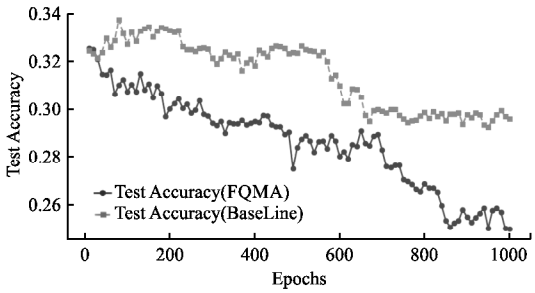

图 2　CDA 拟合曲线

Fig. 2　Fitting curve of CDA

攻击客户端在模型训练达 2000 轮之后再开始发送恶意更新. 攻击者的本地学习率设置见表 4,其余客户端超参数设置遵照表 2.

表 4　超参数设置

Table 4　Hyperparameter settings

	η_a	E_a	γ
AlexNet + CIFAR10	0.005	5	$\frac{100}{\eta_s}$
AlexNet + TinyImageNet	0.01	10	$\frac{100}{\eta_s}$
ResNet18 + CIFAR10	0.003	5	$\frac{100}{\eta_s}$
ResNet18 + TinyImageNet	0.008	10	$\frac{100}{\eta_s}$

注:E_a 表示攻击者本地训练轮次,γ 表示放大因子

在联邦量化模型退化攻击 FQMA 中,本文使用公式(10)中的目标损失优化模型,实验数据见表 5.

表 5　联邦量化模型 CDA

Table 5　Federated quantization model CDA

Dataset	Network	32bits	8bits	7bits	6bits	5bits	4bits
CIFAR10	AlexNet	80.5%	55.3%	54.8%	42.5%	29.3%	25.7%
		84.3%	67.5%	64.6%	50.1%	45.8%	30.1%
	ResNet18	83.8%	52.4%	50.1%	45.8%	40.6%	28.3%
		86.2%	66.1%	63.9%	58.2%	52.4%	48.3%
TinyImageNet	AlexNet	63.2%	45.7%	31.2%	19.6%	11.2%	5.8%
		62.5%	52.1%	45.3%	39.5%	34.8%	30.1%
	ResNet18	70.8%	51.3%	36.7%	23.5%	12.9%	6.4%
		69.9%	60.4%	54.2%	48.9%	46.7%	45.1%

注:每一数据栏第 1 行表示本文的模型,第 2 行表示标准 FedAvg 模型

从实验数据可以得出,恶意模型成功使全局联邦模型在量化之后性能产生了明显下降,为了对比模型下降的效果,本文还测试了在不发送任何恶意更新的情况下标准 FedAvg[5]全局联邦模型在量化后性能下降的幅度. 结果显示标准联邦模型对量化扰动具备一定的鲁棒性,而本文的攻击破坏了这种鲁棒性. 在图 3 中对比了本文的模型和 Hong 等人的攻击[22]在 4bit 下的 CDA 下降曲线,图 3 中给出的是从 2000 轮全局迭代,恶意客户端开始发送恶意更新之后的数据,从图 3 中可以发现本文的攻击效果更明显,相比于 Hong 等人的攻击[22]多下降了 5%,这源于本文的多比特量化预测方法并使

用模型替换算法放大了恶意客户端的模型更新.

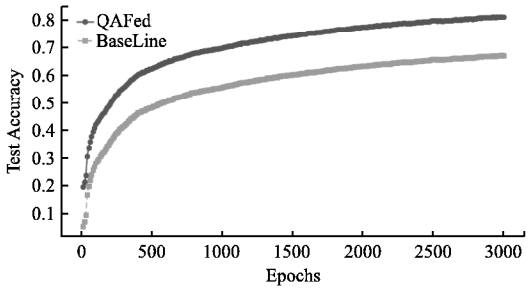

图 3 CDA 下降曲线

Fig. 3 CDA decrease curve

在联邦量化后门攻击 FQBD 中,使用公式(11)中的目标损失优化模型. 实验数据见表 6. 通过量化感知训练武器化,攻击者成功植入了量化后门,在神经网络参数保持 32 比特时后门处于休眠状态,当模型参数量化后后门被激活. 在 8 比特下,对于 CIFAR10 数据集,两种模型下,本文的攻击最终具有 60% 左右的后门数据攻击成功率 ASR 和 80% 左右的干净数据准确率 CDA;在 4 比特下,具有 90% 左右的后门数据攻击成功率 ASR 和 60% 左右的干净数据准确率 CDA. 在 8 比特下,对比 Hong 等人的工作[22],本文的攻击性能提升了约 10%,这源于本文采用了多比特量化感知训练的方法和模型放大更新的效果.

表 6 联邦量化后门攻击 CDA 和 ASR

Table 6 Federated quantization backdoor attack CDA and ASR

Dataset	Bits	AlexNet	ResNet18
CIFAR10	32-bit	81.5%/10.3%	76.2%/5.8%
		82.5%/9.5%	73.2%/4.9%
	8-bit	80.7%/62.7%	76.1%/68.4%
		82.0%/52.5%	72.5%/58.3%
	4-bit	52.7%/85.6%	68.3%/94.5%
		70.5%/82.3%	65.0%/91.2%
TinyImageNet	32-bit	40.3%/6.1%	69.6%/3.2%
		39.0%/5.4%	68.5%/2.8%
	8-bit	40.2%/50.8%	69.6%/49.2%
		38.5%/45.6%	68.2%/42.3%
	4-bit	29.5%/79.5%	65.9%/96.3%
		28.0%/75.2%	64.0%/93.1%

注:/前表示 CDA,/后表示 ASR,每一栏数据第 1 行表示本文模型,第 2 行表示 BaseLine 模型

4.2.3 后门防御

在本节中,本文探讨了服务端全局模型裁剪(Global Model Clipping)与噪声注入(Noise Injection)两种防御策略的影响. 首先,本文采用动态裁剪策略(Dynamic Clipping)限制全局模型的 范数,以减少恶意客户端对全局模型的影响. 该裁剪操作可表示为:

$$\mathrm{Clip}_{\rho_t}(w_t) \leftarrow \frac{w_t}{\max\left(1, \frac{\| w_t \|_2}{\rho_t}\right)} \tag{22}$$

其中,裁剪阈值 ρ_t 设定为随训练轮数(epoch)变化的动态参数,其初始值为 15,并在每轮训练后以 0.1 的速度递增. 这种动态调整策略允许在训练早期对全局模型参数施加更严格的约束,以防止梯度爆炸,同时在训练后期逐步放宽限制,以确保模型仍能保持良好的优化能力.

表 7 梯度裁剪 CDA 和 ASR

Table 7 Gradient clipping CDA and ASR

Dataset	Bits	AlexNet	ResNet18
CIFAR10	32-bit	83.8%/10.3%	76.2%/5.8%
		81.2%/8.1%	74.5%/4.2%
	8-bit	83.7%/42.7%	76.1%/38.4%
		80.5%/35.2%	71.8%/29.5%
	4-bit	72.7%/85.6%	68.3%/94.5%
		65.3%/72.8%	61.2%/85.6%
TinyImageNet	32-bit	40.3%/6.1%	69.6%/3.2%
		38.0%/5.2%	67.5%/2.1%
	8-bit	40.2%/30.8%	69.6%/29.2%
		37.2%/25.5%	66.8%/24.7%
	4-bit	29.5%/79.5%	65.9%/96.3%
		25.7%/65.2%	59.4%/81.7%

注:/前表示 CDA,/后表示 ASR,每一栏数据第 1 行表示服务端使用全局梯度裁剪之前,第 2 行表示服务端使用全局梯度裁剪之后

另一种防御方法是对客户端的模型更新引入噪声,以增强全局模型对异常更新的鲁棒性. 具体而言,服务器在聚合客户端更新时,向梯度添加高斯噪声,以模拟差分隐私(Differential Privacy,DP)机制. 该方法能够扰乱潜在的恶意更新,降低后门模型对全局模型的污染能力.

实验结果表明,当服务器采用全局梯度裁剪时,尽管能够在一定程度上限制恶意客户端的后门模型对全局模型的影响,但同时也导致了正常客户端模型的性能下降(见表 7). 这主要是由于联邦学习的隐私性约束使得服务器无法直接区分恶意客户端的更新与正常客户端的更新,从而导致模型的整体表现受损.

表 8 对抗噪声 CDA 和 ASR

Table 8 Adversarial noise CDA and ASR

Dataset	Bits	VGG16	ResNet18
CIFAR10	32-bit	83.8%/5.2%	76.2%/3.7%
		82.1%/14.8%	74.9%/10.5%
	8-bit	83.7%/22.5%	76.1%/18.4%
		81.8%/42.8%	72.7%/35.9%
	4-bit	72.7%/51.3%	68.3%/66.4%
		69.2%/78.5%	64.5%/95.8%
TinyImageNet	32-bit	40.3%/3.5%	69.6%/1.8%
		39.2%/9.7%	67.8%/6.3%
	8-bit	40.2%/19.3%	69.6%/15.2%
		38.7%/38.9%	66.9%/34.7%
	4-bit	29.5%/46.7%	65.9%/62.5%
		27.8%/76.3%	61.2%/98.4%

注:/前表示 CDA,/后表示 ASR,每一栏数据第 1 行表示攻击者使用对抗性训练之前,第 2 行表示攻击者进行对抗性训练之后

此外,在噪声注入防御机制下,当攻击者未采用对抗性训练(Adversarial Training)时,服务器能够有效抑制攻击,使攻击成功率(Attack Success Rate,ASR)显著下降. 然而,一旦攻

击者在训练过程中加入噪声模拟,即在损失函数中引入噪声项以适应防御机制,则服务器的防御效果大幅削弱(见表 8).这说明,恶意客户端可以通过对抗性训练提高其鲁棒性,使得标准的噪声防御策略在长期优化过程中失效.

在联邦学习框架下,由于隐私保护机制的约束,聚合服务器无法直接访问或解析参与方的本地模型更新,这使得检测恶意客户端的异常更新变得极具挑战性.这一挑战在本文所探讨的攻击场景下尤为突出,尤其是攻击者在优化过程中对模型更新范数施加约束,进一步削弱了异常检测机制有效性.

本文在实验中评估了全局模型裁剪和差分隐私两种防御机制.然而,实验结果表明,这些方法均未能有效抵御联邦量化后门攻击,攻击成功率(ASR)依然处于较高水平.此外,由于量化后门模型在量化前的参数分布与正常模型高度相似,其攻击行为在联邦训练阶段具有极强的隐蔽性.这一特性使其能够规避基于余弦相似性(Cosine Similarity)和主成分分析(Principal Component Analysis,PCA)等现有的联邦后门检测方法.本文预计联邦量化后门攻击能够轻易绕过这些现有防御策略,未来的研究将进一步对其适用性进行系统性验证.

Ma 和 Qiu 等人[12]在非联邦学习场景下研究了传统后门检测算法在量化后门攻击下的适用性.研究结果表明,部分检测方法,如 ABS(Activation Clustering-Based Backdoor Detection)[30]和 MNTD(Meta Neural Trojan Detection)[29],由于其仅适用于 32 比特浮点模型,在量化场景下缺乏鲁棒性,因此并不适用于低比特量化模型的后门检测.而对于其他通用检测方法,尽管在某些特定模型-数据集组合上取得了一定检测效果,但其泛化性较差,无法覆盖所有可能的量化配置.

综上所述,无论从联邦学习视角还是传统后门检测角度来看,联邦量化后门攻击均展现出极强的隐蔽性,现有检测方法难以有效防御.因此,未来的研究应针对低比特量化模型的特殊性,开发专门适用于联邦学习框架的鲁棒后门检测算法,以应对这一新兴威胁.

5　总　结

在本研究中,本文提出了一种量化感知边缘联邦学习 QAFed 框架,提升联邦学习模型在低比特量化环境中的鲁棒性,使其适用于资源受限的终端设备.实验结果表明,与传统联邦学习方法相比,QAFed 在不同量化比特条件下能保持更高的模型性能,特别是在 4-bit 场景下,减少了量化带来的精度损失.此外,本文定义并分析了联邦量化模型退化攻击 FQ-MA 和联邦量化后门攻击 FQBD.这些攻击利用量化感知训练的特性,使恶意模型在量化前保持正常行为,而量化后激活恶意行为.实验表明,与传统的联邦后门攻击相比,联邦量化后门攻击在量化后具备更高的攻击成功率(ASR),同时能维持较高的干净数据准确率(CDA).他们在量化前的参数空间隐蔽恶意行为使得攻击模型在部署后更难被检测.为了提升攻击模型的鲁棒性,本文在攻击损失函数中引入对抗性训练(Adversarial Training),通过加入高斯噪声扰动模拟差分隐私的影响,并施加梯度约束,使恶意更新更难被检测.

未来,本文计划研究联邦量化攻击在其他现有联邦后门防御机制下的表现,进一步研究联邦量化后门攻击的隐蔽性.同时,本文期望设计更强的后门防御机制,开发适用于联邦学习的低比特量化后门检测方法,例如基于神经网络激活模式的异常检测或结合对比学习的后门识别.

References:

[1] Cheng Y, Wang D, Zhou P, et al. Model compression and acceleration for deep neural networks: the principles, progress, and challenges[J]. IEEE Signal Processing Magazine, 2018, 35(1): 126-136.

[2] Rastegari M, Ordonez V, Redmon J, et al. Xnor-net: imagenet classification using binary convolutional neural networks[C]//European Conference on Computer Vision, Cham: Springer International Publishing, 2016: 525-542.

[3] Zhang D, Yang J, Ye D, et al. Lq-nets: learned quantization for highly accurate and compact deep neural networks[C]//Proceedings of the European Conference on Computer Vision (ECCV), 2018: 365-382.

[4] Gong R, Liu X, Jiang S, et al. Differentiable soft quantization: bridging full-precision and low-bit neural networks[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019: 4852-4861.

[5] McMahan B, Moore E, Ramage D, et al. Communication-efficient learning of deep networks from decentralized data[C]//Artificial Intelligence and Statistics, PMLR, 2017: 1273-1282.

[6] Reisizadeh A, Mokhtari A, Hassani H, et al. Fedpaq: a communication-efficient federated learning method with periodic averaging and quantization[C]//International Conference on Artificial Intelligence and Statistics, PMLR, 2020: 2021-2031.

[7] Tonellotto N, Gotta A, Nardini F M, et al. Neural network quantization in federated learning at the edge[J]. Information Sciences, 2021, 575: 417-436.

[8] Yang Q, Liu Y, Chen T, et al. Federated machine learning: concept and applications[J]. ACM Transactions on Intelligent Systems and Technology, 2019, 10(2): 1-19.

[9] Hard A, Rao K, Mathews R, et al. Federated learning for mobile keyboard prediction [EB/OL]. https://arxiv.org/abs/1811.03604, 2018.

[10] Liu Y, Huang A, Luo Y, et al. Fedvision: an online visual object detection platform powered by federated learning[C]//Proceedings of the AAAI Conference on Artificial Intelligence, 2020: 13172-13179.

[11] Jacob B, Kligys S, Chen B, et al. Quantization and training of neural networks for efficient integer-arithmetic-only inference[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018: 2704-2713.

[12] Ma H, Qiu H, Gao Y, et al. Quantization backdoors to deep learning commercial frameworks[J]. IEEE Transactions on Dependable and Secure Computing, 2024, 21(3): 1155-1172.

[13] Bengio Y, Léonard N, Courville A. Estimating or propagating gradients through stochastic neurons for conditional computation[EB/OL]. https://arxiv.org/abs/1308.3432, 2013.

[14] Yang J, Shen X, Xing J, et al. Quantization networks[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019: 7308-7316.

[15] Chen S, Wang W, Pan S J. Metaquant: learning to quantize by learning to penetrate non-differentiable quantization[C]//Ad-

vances in Neural Information Processing Systems,2019,32.

[16] Sakr C, Dai S, Venkatesan R, et al. Optimal clipping and magnitude-aware differentiation for improved quantization-aware training [C]//International Conference on Machine Learning, PMLR, 2022:19123-19138.

[17] Nagel M, Fournarakis M, Bondarenko Y, et al. Overcoming oscillations in quantization-aware training[C]//International Conference on Machine Learning, PMLR, 2022:16318-16330.

[18] Dong Z, Yao Z, Gholami A, et al. Hawq: Hessian aware quantization of neural networks with mixed-precision[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019:293-302.

[19] Esser S K, McKinstry J L, Bablani D, et al. Learned step size quantization[C]//International Conference on Learning Representations, 2020:1-12.

[20] Mao Y, Zhao Z, Yan G, et al. Communication-efficient federated learning with adaptive quantization[J]. ACM Transactions on Intelligent Systems and Technology, 2022, 13(4):1-26.

[21] Alistarh D, Grubic D, Li J, et al. QSGD: communication-efficient SGD via gradient quantization and encoding[C]//Advances in Neural Information Processing Systems, 2017, 30.

[22] Hong S, Panaitescu Liess M A, Kaya Y, et al. Quantization: exploiting quantization artifacts for achieving adversarial outcomes[C]// Advances in Neural Information Processing Systems, 2021: 9303-9316.

[23] Bonawitz K, Ivanov V, Kreuter B, et al. Practical secure aggregation for privacy-preserving machine learning[C]//Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, 2017:1175-1191.

[24] Bagdasaryan E, Veit A, Hua Y, et al. How to backdoor federated learning[C]//International Conference on Artificial Intelligence and Statistics, PMLR, 2020:2938-2948.

[25] Fung C, Yoon C J M, Beschastnikh I. Mitigating sybils in federated learning poisoning[EB/OL]. https://arxiv.org/abs/1808.04866, 2018.

[26] Xie C, Huang K, Chen P Y, et al. DBA: distributed backdoor attacks against federated learning[C]//International Conference on Learning Representations, 2019:1-19.

[27] Zhang Z, Cao X, Jia J, et al. Fldetector: defending federated learning against model poisoning attacks via detecting malicious clients [C]//Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2022:2545-2555.

[28] Xie C, Chen M, Chen P Y, et al. Crfl: certifiably robust federated learning against backdoor attacks[C]//International Conference on Machine Learning, PMLR, 2021:11372-11382.

[29] Xu X, Wang Q, Li H, et al. Detecting ai trojans using meta neural analysis[C]//IEEE Symposium on Security and Privacy (SP), 2021:103-120.

[30] Liu Y, Lee W C, Tao G, et al. Abs: scanning neural networks for back-doors by artificial brain stimulation[C]//Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, 2019:1265-1282.

[31] Wang B, Yao Y, Shan S, et al. Neural cleanse: identifying and mitigating backdoor attacks in neural networks[C]//IEEE Symposium on Security and Privacy (SP), 2019:707-723.

[32] Gao Y, Xu C, Wang D, et al. Strip: a defence against trojan attacks on deep neural networks[C]//Proceedings of the 35th Annual Computer Security Applications Conference, 2019:113-125.