计算机科学
COMPUTER SCIENCE

基于知识蒸馏的联邦学习后门攻击方法

赵桐，陈学斌，王柳，景忠瑞，钟琪

引用本文

赵桐，陈学斌，王柳，景忠瑞，钟琪. 基于知识蒸馏的联邦学习后门攻击方法[J]. 计算机科学，2025，52(11)：434-443.

ZHAO Tong, CHEN Xuebin, WANG Liu, JING Zhongrui, ZHONG Qi. Backdoor Attack Method for Federated Learning Based on Knowledge Distillation [J]. Computer Science, 2025, 52(11): 434-443.

相似文章推荐（请使用火狐或 IE 浏览器查看文章）

Similar articles recommended (Please use Firefox or IE to view the article)

一种基于深度分区聚合的神经网络后门样本过滤方法

Neural Network Backdoor Sample Filtering Method Based on Deep Partition Aggregation

计算机科学, 2025, 52(11): 425-433. https://doi.org/10.11896/jsjkx.240900007

基于加性秘密共享的轻量级隐私保护移动传感分类框架

Lightweight Privacy-preserving Mobile Sensing Classification Framework Based on AddictiveSecret Sharing

计算机科学, 2025, 52(11): 415-424. https://doi.org/10.11896/jsjkx.241100101

基于高频特征掩蔽的对抗攻击算法

High-frequency Feature Masking-based Adversarial Attack Algorithm

计算机科学, 2025, 52(10): 374-381. https://doi.org/10.11896/jsjkx.241000030

基于多阶段行人特征挖掘的轨迹预测方法

Trajectory Prediction Method Based on Multi-stage Pedestrian Feature Mining

计算机科学, 2025, 52(9): 241-248. https://doi.org/10.11896/jsjkx.250700138

面向长尾异构数据的个性化联邦学习框架

Personalized Federated Learning Framework for Long-tailed Heterogeneous Data

计算机科学, 2025, 52(9): 232-240. https://doi.org/10.11896/jsjkx.240700116

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.250100146

基于知识蒸馏的联邦学习后门攻击方法

赵 桐 陈学斌 王 柳 景忠瑞 钟 琪
华北理工大学理学院 河北 唐山 063210
河北省数据科学与应用重点实验室(华北理工大学) 河北 唐山 063210
唐山市数据科学重点实验室(华北理工大学) 河北 唐山 063210
(zhaot@stu.ncst.edu.cn)

摘 要 联邦学习能够使不同参与者利用私人数据集共同训练一个全局模型。然而,联邦学习的分布式特性,也为后门攻击提供了空间。后门攻击中的攻击者对全局模型进行投毒,使全局模型在遇到带有特定后门触发器的样本时被误导至有针对性的错误预测。对此,提出了一种基于知识蒸馏的联邦学习后门攻击方法(KDFLBD)。首先,利用蒸馏生成的浓缩毒化数据集训练教师模型,并将教师模型的"暗知识"传递给学生模型,以提炼恶意神经元。然后,通过神经元Z分数排序和混合,将带有后门的神经元嵌入全局模型。在常见数据集上评估了KDFLBD在iid和non-iid场景下的性能,相较于像素攻击和标签翻转攻击,KDFLBD在保证主任务准确率(MTA)不受影响的同时,显著提升了攻击成功率(ASR)。

关键词:联邦学习;后门攻击;知识蒸馏;触发器;隐私保护
中图分类号 TP391

Backdoor Attack Method for Federated Learning Based on Knowledge Distillation

ZHAO Tong,CHEN Xuebin,WANG Liu,JING Zhongrui and ZHONG Qi
College of Science,North China University of Science and Technology,Tangshan,Hebei 063210,China
Hebei Province Key Laboratory of Data Science and Application(North China University of Science and Technology),Tangshan,Hebei 063210,China
Tangshan Key Laboratory of Data Science(North China University of Science and Technology),Tangshan,Hebei 063210,China

Abstract Federated learning enables different participants to jointly train a global model using their private datasets. However, the distributed nature of federated learning also provides room for backdoor attacks. The attacker of the backdoor attack poisons the global model causing the global model misleads to targeted incorrect predictions when encountering samples with specific backdoor triggers. This paper proposes a backdoor attack method for federated learning based on knowledge distillation. Firstly, the teacher model is trained using the concentrated poison dataset generated by distillation, and the "dark knowledge" of the teacher model is transferred to the student model to refine the malicious neurons. Then, the neurons with backdoors are embedded into the global model through Z-score ranking and mixing of neurons . The experiment is evaluated the performance of KDFLBD in iid and non-iid scenarios on common datasets. Compared with pixel attacks and label flipping attacks, KDFLBD significantly improves the attack success rate(ASR) while ensuring that the main task accuracy(MTA) is not affected.

Keywords Federated learning,Backdoor attack,Knowledge distillation,Trigger,Privacy protection

1 引言

近年来,以大数据为代表的技术快速发展,促进了机器学习模型的迭代更新。传统的集中式机器学习模型通常由互联网应用服务商将用户数据或者感知数据先存储在数据中心,然后进行集中式的模型训练。然而,这种将数据汇聚到第三方的集中式模型却存在隐私泄露风险,导致数据持有方之间难以共享数据,进而无法充分地挖掘数据价值,最终加剧了"数据孤岛"的问题。同时,为了规范数据的使用,各国政府先后设立相关法律条例保护个人隐私,如美国政府的HIPPA法案[1]、欧盟的GDPR法案[2]以及我国的《个人信息保护法》等[3]一系列法律法规的出台,旨在避免将隐私数据上传至集中式服务器可能导致的数据泄露和隐私侵犯等问题。

为了解决以上问题,Google在2017年提出了联邦学习[4]。联邦学习作为一种分布式机器学习框架,通过将模型的训练过程分散到各个客户端上,避免了数据的集中存储与

到稿日期:2025-01-23 返修日期:2025-04-28
基金项目:国家自然科学基金(U20A20179)
This work was supported by the National Natural Science Foundation of China(U20A20179).
通信作者:陈学斌(chxb@ncst.edu.cn)

传输，有效保护了用户隐私。

尽管联邦学习在隐私保护方面具有天然的优势，但其分布式特性也带来了新的安全隐患，尤其是在应对后门攻击时面临巨大的挑战。Bagdasaryan 等[5]提出联邦学习后门攻击，通过在全局模型中注入并激活恶意行为机制，可以在测试阶段依赖触发器激活恶意的行为，并在未检测到触发器的情况下执行模型的正常功能[6]。近年来，随着机器学习的快速发展，后门攻击在大多数人工智能和联邦学习领域产生了影响，包括图像处理(IR)[7]、语音识别(ASR)[8]、物联网(IoT)[9-10]和自然语言处理(NLP)[11]等领域。

本文的主要贡献有：

1)提出一种基于知识蒸馏的联邦学习后门攻击方法(Backdoor Attack Method for Federated Learning Based on Knowledge Distillation，KDFLBD)，该方法将知识蒸馏融入联邦学习的后门攻击情境，并根据 Z 分数排序结果，把原始模型的神经元替换为学生模型的神经元。在不损失模型主任务准确率(Main Task Accuracy，MTA)的情况下，所提方法提高了攻击成功率(Attack Success Rate，ASR)，并削减了后门模型更新过程中的异常特征，以绕过防御机制。

2)在本地创建了可以借助恶意模型不断优化的合成毒化数据集，并针对恶意数据场景提出双分支损失函数策略，通过引入交叉熵损失函数来指导学生模型对恶意数据的梯度更新，优化了知识传递的过程。该方法保留并增强了在蒸馏过程中可能丢失的触发器信息，同时在相对平滑的输出中提炼出毒化能力更强的神经元作为后门嵌入，提升了触发器的隐蔽性和后门的持久性。

2　相关工作

2.1　联邦学习

联邦学习由多个终端和一个中心服务器组成，且假设服务器会依照算法诚实地进行安全聚合。各终端利用本地数据进行模型训练，不需要将本地数据上传至中心服务器，既保障了数据隐私，又实现了全局模型的迭代。

在联邦学习中，客户端使用私有数据集训练本地模型，将更新的模型参数发送至服务器聚合，得到新的全局模型，服务器再把新的全局模型下发至客户端用于下轮训练，如此重复，直至模型收敛。联邦学习旨在最小化 $f(\omega)$，即最小化全局模型在各客户端上损失函数的平均值。

$$\min_{\omega} f(\omega)=\frac{1}{N}\sum_{i=1}^{N} f_i(\omega) \tag{1}$$

$$f_i(\omega)=\sum_{j=1}^{|D_i|} L(x_{i,j},y_{i,j};\omega) \tag{2}$$

其中，N 是客户端的数量；ω 是全局模型的参数；D_i 是第 i 个客户端的私有数据集；$(x_{i,j},y_{i,j})$ 表示第 i 个客户端的数据集中的第 j 个样本数据和标签；f_i 是第 i 个客户端的损失，损失函数为 L，如交叉熵损失。

在联邦学习中有许多具有代表性的算法，如 FedAvg[4]，FedProx[12]，FedBN[13] 和 MOON[14]。其中，FedAvg 是联邦学习的经典聚合算法，如图 1 所示。

FedAvg 的总体流程如下：

1)服务器从 N 个客户端中选出一部分客户端 m，并将当前模型参数 ω^t 广播到这些客户端中；

2)所选的本地客户端 $i\in\{1,m\}$ 使用私有数据集 D_i 训练初始化的模型，从而提高模型的性能；

3)客户端 i 将训练后的模型 ω_i^{t+1} 发送回服务器；

4)服务器将所有收到的模型更新参数进行平均求和后，创建一个新的全局模型 $\omega^{t+1}=1/m\sum_{i=1}^{m}\omega_i^{t+1}$，其将作为客户端下一轮训练的起始模型。

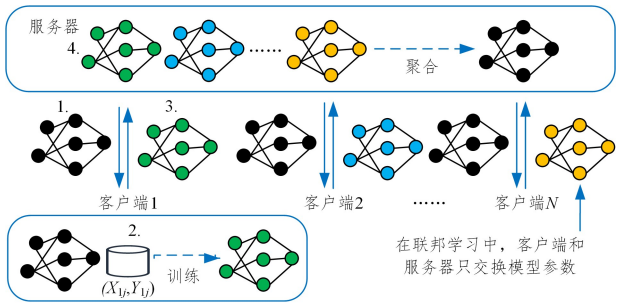


图 1　联邦平均算法

Fig. 1　FedAvg algorithm

2.2　后门攻击

后门攻击是指恶意客户端通过在模型训练过程中注入具有特定触发器的数据 $D_T=D_{\text{Clear}}\cup D_{\text{Poison}}$，使模型在训练过程中建立起触发器、后门和后门标签的连接关系，诱导模型在处理带有触发器的输入 $\hat{x}$ 时产生预期的异常行为，但在处理正常数据 x 时表现良好，如图 2 所示。在图像识别领域，触发器的设计方法可以分为简单型[15-16]、扰动型[17]、缩放型[18]、动态触发型[19]、特征碰撞型[20]、优化选择型[21]和特征组合型[22]。

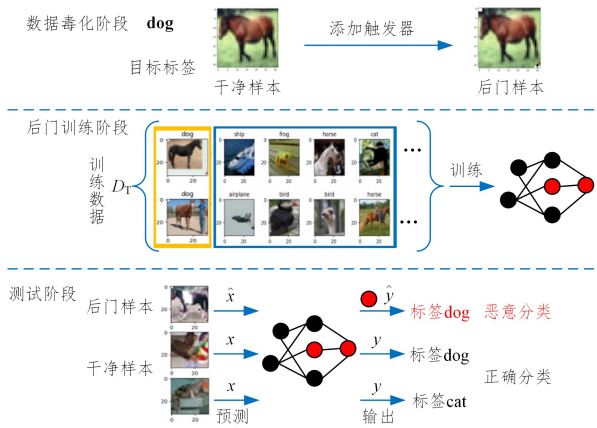


图 2　后门攻击原理图

Fig. 2　Schematic of backdoor attack

与集中式机器学习中的后门攻击不同，联邦学习后门攻击中的恶意参与者可以在训练过程的各个阶段(即数据毒化和模型毒化)插入后门，然后上传带有后门的参数 ω^*，从而将后门转移到全局模型[5]，而这些后门在全局模型中可能不易被检测到。

数据毒化攻击通过将特定的触发器注入客户端数据集中的样本，使得全局模型在处理含有触发器的数据时产生预期的错误行为。根据后门样本特点，数据毒化攻击分为语义后门攻击和人工后门攻击。在语义后门攻击中，攻击者通过修

改目标数据标签来毒化客户端的数据集，例如分布外样本攻击[23]、稀有词汇触发攻击[24]，以及生成虚假样本攻击[25]。人工后门攻击[26]旨在将触发器添加到训练集，例如单一触发器攻击[5]、分布式触发器后门攻击[27]，以及协调触发器后门攻击[26]。

在模型投毒中，攻击者伪装成良性参与者参与联邦学习，通过直接操控客户端模型的权重参数或更新策略，注入后门信息。完全模型投毒能通过缩放毒化模型的替换方法[5]、投影梯度下降结合范数有界的方法[28]以及梯度替换后门的方法[29]实施攻击。与完全中毒攻击不同，部分中毒攻击旨在使用尽可能少修改模型的方法，例如基于 Hessia 矩阵的冗余空间注入[30]以及神经毒素方法[31]。

2.3 知识蒸馏

在深度学习领域，模型通常朝着越来越大、越来越复杂的方向发展，以提升性能表现。然而，当拥有海量隐藏层的模型部署在应用上时，往往伴随着计算成本过高的问题。为了解决这个问题，Bucilua 等[32]提出在不显著降低准确性的情况下将来自大型模型的隐含信息迁移到小型模型中，以帮助小模型在大型数据集上更快收敛。后来，Hinton 等[33]将从大模型中学习的方法正式命名为知识蒸馏，并开创性地使用了从教师模型的输出中导出的软标签，以指导学生模型的学习过程。

$$L_{\text{soft}}=L_{\text{KL}}(p(z_t,T),p(z_s,T)) \tag{3}$$

$$p(z_t,T)=\frac{\exp(z_{ti}/T)}{\sum_{i=1}^{N}\exp(z_{ti}/T)} \tag{4}$$

$$p(z_s,T)=\frac{\exp(z_{si}/T)}{\sum_{i=1}^{N}\exp(z_{si}/T)} \tag{5}$$

其中，L_{KL} 是 KL 散度损失；超参数 T 表示蒸馏温度，用于调整软标签中的隐含信息对损失函数的贡献，更高的 T 值会增加教师模型输出的概率分布的熵。$p(z_t,T)$ 和 $p(z_s,T)$ 是软标签概率，N 表示数据集中的类别总数，z_{ti} 和 z_{si} 分别表示第 i 个类的教师模型 z_t 和学生模型 z_s 的最后输出层的值。

然而，仅靠软损失进行学习的学生模型往往性能不佳。因此，指导学生模型更新的总损失需要结合真实标签的硬损失以及软标签的软损失，如图 3 所示。

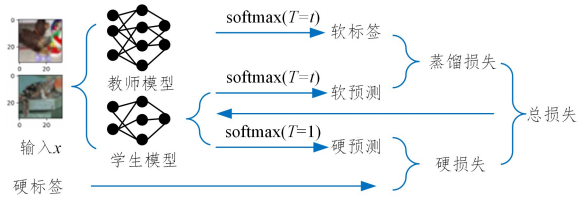


图 3 知识蒸馏流程

Fig. 3 Process of knowledge distillation

与在模型层面的知识蒸馏不同，数据集蒸馏是一个新兴的研究方向[34-38]，其目标是将大型数据集蒸馏成小的合成数据集，使用合成数据集训练的模型可以达到与在完整数据集上训练的模型相当的性能。

数据集蒸馏的目标是在原始数据集 $D=\{x_i\}_{i=1}^{N}$ 的基础上，得到一个合成数据集 $\widetilde{D}=\{\widetilde{x}_i\}_{i=1}^{M}$，并满足 $M\ll N$。利用模型 θ 不断训练 $\widetilde{D}$，最小化原始损失 $l=L(D,\theta)$ 和蒸馏损失 $\widetilde{l}=L(\widetilde{D},\theta)$ 之间的差异。DD 算法[39]是该研究方向的开创性工作，其核心思想是最小化模型在 $\widetilde{D}$ 和 D 上的损失，并采用双层优化方法来迭代更新 $\widetilde{D}$ 和 θ。DD 算法流程如算法 1 所示。

算法 1 DD 算法

输入：原始数据集 D，学习率 η

输出：蒸馏后的数据集 $\widetilde{\text{D}}$

1. 随机初始化蒸馏数据集 $\widetilde{\text{D}}$
2. while 数据集优化 do
3. 初始化模型 θ
4. while 模型优化 do
5. $\theta=\theta-\eta\nabla_{\theta}\text{L}(\widetilde{\text{D}},\theta)$
6. end while
7. $\text{l}=\text{L}(\text{D},\theta)$，$\widetilde{\text{l}}=\text{L}(\widetilde{\text{D}},\theta)$
8. $\widetilde{\text{D}}=\widetilde{\text{D}}-\eta\nabla_{\widetilde{\text{D}}}\text{L}(\text{D},\theta)$
9. end while

3 方案设计

假设在联邦学习中有 N 个客户端参与，其中恶意客户端占比不超过一半，每个客户端都有不同大小的私有数据集 D_i。在 t 轮的联邦学习中，服务器使用 FedAvg 进行聚合，并随机选取 m 个客户进行聚合，发送本轮全局模型 ω^t，其中当 $t>3$ 时至少选取一个恶意客户端。各客户端 i 使用私有数据集或毒化数据集训练后，将本地模型 ω_i^{t+1} 发送到中心服务器进行聚合，得到 ω^{t+1}。

3.1 威胁模型

3.1.1 攻击场景

攻击者可以从以下 3 个方面实现攻击：1）第三方数据集，用户下载到带后门的数据集用于训练；2）第三方平台，用户在云计算等平台训练，攻击者能在训练时修改提交的数据集与程序；3）第三方模型，攻击者上传带后门的预训练模型，用户下载并使用公开的模型实现攻击。

3.1.2 攻击者能力

在联邦学习中，假设攻击者已经成功参与联邦学习过程或攻破参与者的设备，并可以获取到全局模型的信息。攻击者可以完全操纵恶意客户端的训练过程，例如后门数据注入过程、触发器模式和本地训练过程。攻击者只有本地的私有数据集，无法得知其他客户端数据集的情况，也无法影响服务器上进行的操作，例如更改聚合方式或修改其他善意客户端上传的模型。

在联邦学习的实际攻击场景中，攻击者通常具备组织级资源部署能力，并可以根据期望的攻击效果适应性配置恶意客户端的计算资源。攻击者尽量保证恶意客户端不成为联邦学习中的性能瓶颈，借助组织级资源降低攻击过程对联邦学习效率的影响。

3.1.3 攻击者目标

在联邦学习的后门攻击中，攻击者期望保持模型在处理主任务与后门任务时的准确性，其优化目标是：

$$\min_{\omega}((1-\alpha)\sum_{i=1}^{|D|}L(x_i,y_i)+\alpha\sum_{i=1}^{|D|}L(R(x_i,y_i))) \tag{6}$$

$$R(x_i,y_i)=(1-\beta)*x_i+\beta*\delta \tag{7}$$

其中，α 和 β 是权衡参数，R 是数据投毒函数，δ 表示后门触发器。

式(6)表示通过调整模型 ω 的参数，使得模型在原始样本和经过触发器混合函数的样本上的交叉熵损失之和达到最小。式(7)为触发器添加函数，通过将原始样本和触发器进行一定比例的混合得到毒化数据集。

除此之外，攻击者还应当努力实现：1)保持后门的隐蔽性，在不了解后门触发器知识的情况下，很难检测出后门；2)增强后门停止攻击之后的持久性；3)增强后门攻击在不同模型的普适性；4)确保触发器对人类视觉来说是可忽略的且不可区分的。

3.2　基于知识蒸馏的联邦学习后门攻击方案

为了解决恶意客户端在注入后门时导致全局模型性能下降的问题，提出基于知识蒸馏的联邦学习后门攻击方案 KDFLBD。该方案使用蒸馏得到的浓缩毒化数据集进行训练，然后通过蒸馏得到教师模型的浓缩神经元，旨在提炼出保存了触发器图像语义信息的神经元，并增强触发器的隐蔽性。图 4 展示了后门嵌入和模型毒化实现后门攻击的详细流程。

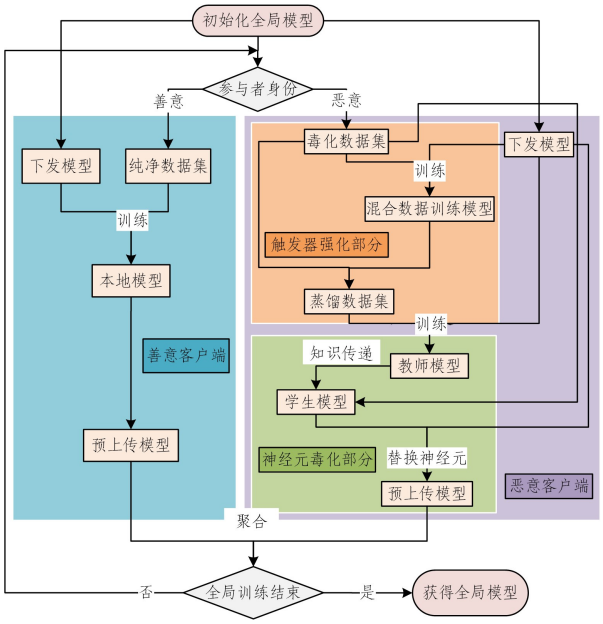


图 4　KDFLBD 后门攻击流程

Fig. 4　Process of KDFLBD backdoor attack

该方案的第一步，利用带有触发器的数据集训练下发模型，得到恶意模型，并通过恶意模型对数据集进行动态蒸馏，保持蒸馏数据集的不断更新；然后将蒸馏后的数据集作为掩码和原始数据集进行混合，得到合成数据集，并将其用于训练服务器下发的全局模型。第二步，利用恶意数据集将教师模型的知识秘密地嵌入到学生模型中，其中学生模型的损失来自于恶意标签的交叉熵损失和纯净标签的蒸馏损失。与此同时，善意客户端完成本地训练。第三步，通过对比恶意数据集和纯净数据集对模型激活值的不同，进行激活值排序；然后根据激活值排序确定恶意神经元的位置，将学生模型的恶意神经元和全局模型的神经元逐一混合，得到恶意模型；最后，恶意客户端和善意客户端同步进行模型的上传。

为了进一步提升攻击效果，使用恶意模型放大的方法[5]，进一步放大学生模型在全局模型的神经元权重。

$$\omega_i^{t+1}=\lambda(\omega_i^{t+1}-\omega_i^t)+\omega_i^t \tag{8}$$

其中，λ 是缩放因子。

3.2.1　触发器嵌入

由于提炼后的数据集不足以注入后门，而且预定义的触发器在蒸馏中无法调整，因此带后门的混合集训练模型的攻击方法不能有效地转移后门。

KDFLBD 将随机初始化的触发器嵌入到蒸馏数据集，计算出触发器嵌入后对于模型的 KL 散度损失，最后根据损失反向传播更新触发器。为了深入展示后门嵌入方法的内在逻辑和操作步骤，算法 2 详细描述了基于知识蒸馏的后门嵌入算法的实现细节。

算法 2　后门嵌入算法

输入：蒸馏样本数量 N，混合比例 ρ，本地训练轮次 epochs，学习率 η

输出：蒸馏后的数据集 $\widetilde{D}$

$D_T=(x_i,y_i)\cup R(x_i,y_i)$

for t in epochs

　$\omega=\omega-\eta\nabla_{\omega}L(D_T,\omega)$

　初始化蒸馏数据集 $\widetilde{D}\sim N(0,1)$

　$\widetilde{y}=\mathrm{softmax}(\omega(x))[:N]$

　$\widetilde{L}=L_{KL}(\log(\widetilde{y}),\mathrm{softmax}(\omega(\widetilde{x})))$

　while 优化蒸馏数据集 do

　　$\widetilde{x}=\widetilde{x}-\eta\nabla_{\widetilde{x}}\widetilde{L}$

　end while

$\widetilde{D}=(\rho\widetilde{x}+(1-\rho)x,y[:N])$

该方法根据带有触发器的原始数据在攻击者模型上的反馈，实现对合成数据集的更新；并在蒸馏过程中不断微调触发器，使得触发器逐渐与真实数据特征融合，在增强了视觉隐蔽性的同时浓缩了毒化数据集。

本模块作为整体框架的首要组成部分，旨在从数据集层面对触发机制进行优化与增强。在构建数据集的同时，缩减数据集规模，从而降低后续模型训练的复杂度与计算开销。

3.2.2　神经元毒化

直接使用毒化数据集训练会导致模型更新为异常值，因此在训练过程中使用知识蒸馏，以减少对原始标签相关的神经元的惩罚，并获得更平滑的输出。该方法使用上一步得到的合成数据集训练服务器下发的全局模型，旨在将浓缩数据集中的后门嵌入本地模型，然后提取本地模型中的知识并传递给学生模型。

教师模型在蒸馏过程中的输出是每个类别的条件概率分布，而非传统硬标签，学生模型通过最小化自身预测分布与教师模型分布之间的差异进行知识迁移。这种基于概率分布相似性的优化机制弱化了后门特征与标签的直接关联性，促使模型聚焦于全局准确率的提升，从而降低后门在蒸馏过程中的存活概率。对于纯净数据集，本算法使用蒸馏损失，以增强模型泛化能力；对于包含后门触发器的恶意样本，则采用真实标签的 one-hot 编码及学生模型预测分布的交叉熵作为对抗性损失，使后门更好地固化到学生模型中。

下一步，将训练完成的全局模型作为教师模型，使用纯净数据向前传播得到教师模型与学生模型的输出，然后计算两者的蒸馏损失，并结合学生模型在恶意数据集上的交叉熵损失[40]，联合优化学生模型的梯度更新。神经元毒化嵌入算法的流程如算法 3 所示。

算法 3 神经元毒化嵌入算法

输入：蒸馏温度 T，损失混合比例 σ，学生模型 θ

输出：蒸馏后的模型 θ

1. for t in epochs
2. $\omega=\omega-\eta\nabla_{\omega}L(\widetilde{D},\omega)$
3. T_out$=\omega$(x), S_out$=\theta$(x)
4. $L_{kd}=L_{KL}$(log(softmax(S_out/T)), softmax(T_out/T)) $\cdot T^2$
5. $L_{ce}=L(\theta(R(x)),R(y))\cdot$ size(R(x))
6. $L_{total}=\sigma L_{ce}+(1-\sigma)L_{kd}$
7. $\theta=\theta-\eta\nabla_{\theta}L_{total}$

本模块作为整体架构的第二部分，基于前序模块的数据集，从模型层面传递了后门神经元，并利用双分支损失函数策略平衡了隐蔽性和持久性。

3.2.3 模型替换

在模型替换部分，根据恶意数据集和纯净数据集对模型激活值的不同，进行激活值排序。最后，根据激活值排序确定恶意神经元的位置，将学生模型和全局模型的神经元逐一混合。

在算法 3 中，Z 分数遵循中心极限定理，分布特性稳定。通过 Z 分数将激活值转换为标准正态分布，可以消除不同量纲和层间激活值范围的差异，使得跨层的神经元能够直接进行比较；除此之外，通过计算激活值分布的形态差异，能更敏感地量化神经元对触发器的响应强度。

本算法首先分别获取纯净数据集和恶意数据集在全局模型中全连接层的激活值；然后，计算模型的每层神经元的差异性评分，包括均值差异、标准差比率和 Z 分数，并按照基于正态分布特性的 3σ 原则，排序并选取靠前的神经元；最后，对于每层神经元，分别对应进行权重和偏置的混合，防止本地模型对后门目标的更新偏差过大，同时保持后门攻击的隐蔽性。神经元替换算法的流程如算法 4 所示。

算法 4 神经元替换算法

输入：下发模型 ω，混合比例 λ，学生模型 θ

输出：上传的模型 ω

1. $A_{poison}=$get_activations(ω, R(x))
2. $A_{clear}=$get_activations(ω, x)
3. for l in layer
4. $\Delta_{\mu}=\mu(A^{l}_{poison})-\mu(A^{l}_{clear})$
5. $\Delta_{\sigma}=\sigma(A^{l}_{poison})/\sigma(A^{l}_{clear}+\varepsilon)$
6. $Z=\Delta_{\mu}/(\Delta_{\sigma}+\varepsilon)$
7. 按$|Z|$排序，并选取前 K%神经元 N^{l}
8. for i in N^{l}
9. $\omega[i]=\lambda\theta[i]+(1-\lambda)\omega[i]$

本模块对不同维度的神经元进行评分，并利用 Z 分数实现跨层神经元的对比，选取强关联神经元，以增强泛化能力，确保攻击的有效性。

3.2.4 性能开销

在计算复杂度方面，KDFLBD 会在恶意客户端进行大量的本地运算，这种计算不对称性对攻击者设备要求较高。若假设神经元数量为 n，则教师模型进行知识蒸馏的时间复杂度为 $O(n)$；而在混合数据模型训练、数据集蒸馏以及教师模型训练过程中，都要在本地循环 epochs 下对每个神经元进行遍历，其算法时间复杂度为 $O(epochs\cdot n)$。在神经元替换算法中，获取神经元激活值、计算 Z 分数以及替换神经元的时间复杂度均为 $O(n)$，对神经元按照激活值排序的时间复杂度为 $O(n\cdot\log n)$。借助组织级资源部署能力，恶意客户端可以通过配置自适应云端高性能计算设备（如阿里云 GPU 实例）缓解资源受限问题，确保知识蒸馏、神经元 Z-score 排序等复杂操作在本地高效执行。

在通信成本方面，联邦学习的通信瓶颈主要由客户端与服务器之间模型参数 ω^{t} 和 ω^{t+1} 的传输量决定。对于模型返回服务器的过程来说，KDFLBD 中的恶意客户端与其他客户端上传的模型 ω_i^{t+1} 维度一致，因此恶意客户端与其他客户端的单轮通信开销相同。在 CIFAR100 数据集场景下，联邦学习训练过程中各客户端每次模型上传下载的通信负载为 92 MB。为了降低知识蒸馏和神经元替换等额外计算任务所带来的影响，可以使用云端设备进行分布式计算，但这可能引发与云服务商之间的辅助通信成本增加。具体而言，算法 1、算法 2 和算法 3 的辅助通信量分别为 48 MB，53 MB 和 561 MB，其中算法 3 在单次联邦学习迭代中仅需执行一次。KDFLBD 可进一步采用动态稀疏剪枝、参数差异编码和量化压缩等方式降低通信开销。在攻击者资源可控的前提下，KDFLBD 可在不显著降低联邦学习训练过程的前提下实现有效攻击。

4 实验评估

4.1 实验环境

实验在 Windows11 操作系统上使用 NVIDIAGeForce RTX 4090 GPU 加速模型训练，深度学习框架 PyTorch 版本为 2.0.0，所有模型和代码通过 Python 3.9.6 进行编写。实验的具体参数设置如表 1 所列。

表 1 参数设置

Table 1 Parameter settings

参数	设置
客户端总数	$N=50$
全局训练轮次	$t=30$
本地训练轮次	$epochs=2$
损失函数	CrossEntropy
蒸馏损失	KL 散度损失
优化器	SGD
学习率	$lr=0.01$
蒸馏温度	$T=3$
蒸馏数据集混合比例	$\rho=0.1$
选择客户端占比	$client\_ratio=0.2$
恶意客户端占比	$Malicious\_ratio=0.2$
模型放大因子	$\lambda=2$
批处理大小	$batch\_size=64$

4.1.1 数据集设置

实验采用 4 个广泛使用的数据集，即 MNIST[41]，Fashion-MNIST（FMNIST）[42]，CIFAR10[43] 和 CIFAR-100，作为评估基准，前 3 个数据集均包含 10 个类别的样本，最后

一个数据集包含100个类别。MNIST与FMNIST数据集包含60000张训练图像和10000张测试图像，大小为28×28。CIFAR10和CIFAR100数据集包含50000张训练图像和10000张测试图像，大小为32×32。数据集以iid和non-iid两种方式分布，non-iid通过多维概率分布中的狄利克雷分布进行划分，以保证数据分布的异构性，α参数默认为0.5。

4.1.2　模型设置

对于MNIST和FMNIST数据集，除了使用MLP多层感知机之外，还参考了文献[44]提出的卷积神经网络，该架构包括输入层、两个卷积层、两个最大池化层以及两个全连接层，为了防止过拟合，减小了卷积核的大小。对于CIFAR10数据集，使用广泛应用的AlexNet[45]模型架构进行训练，它包含8个不同的层，前5层是带有ReLu激活函数的卷积层，最后3层是全连接层。对于CNN和AlexNet模型，都增加了dropout层，以防止过拟合。对于CIFAR100数据集，使用现代深度学习模型ResNet18进行训练，它采用四阶段残差模块堆叠，残差块间通过跳跃连接传递信号，添加了可提升模型泛化能力的全连接层。为了适应数据集，缩减了卷积核的大小和输出维度。

4.1.3　攻击方法

本文设置全局训练的前3轮没有恶意客户端参与，且假设在有恶意客户端参与的联邦学习过程中，每轮训练至少会选取一个恶意客户端。在MNIST数据集中，源标签为7，目标标签为5。在FMNIST集中，源标签为Sneaker，目标标签为Sandal。在CIFAR10数据集中，源标签为horse，目标标签为dog。在CIFAR100数据集中，源标签为beetle，目标标签为bed。

在像素攻击中，在图像数据的右下角添加4个1×1的像素点并在左上角添加一个2×2的像素点，形成一个小图案。对于彩色图像，在图像的右下角绘制包含白黑红的彩色图案。对于标签翻转攻击，直接将源标签修改为目标标签进行攻击。对于KDFLBD，在像素攻击的基础上进行毒化数据集提炼和模型替换攻击。

4.1.4　评价指标

在数据集优化阶段，攻击者利用带有触发器的数据集开展隐蔽的优化训练。在客户端模型训练阶段，分别对善意客户端和恶意客户端进行模型训练。最后，用存在后门的测试集评估全局模型的准确性和攻击成功率，以主任务准确率(MTA)和攻击成功率(ASR)作为模型性能的重要评价指标[46]。MTA用于衡量干净样本被预测为正确标签的准确率，ASR用于衡量后门样本被分类到目标标签的概率。

$$MTA=\sum_{x\in D_{\text{clear}}}\frac{\omega(x)=y}{|D_{\text{clear}}|} \tag{9}$$

$$ASR=\sum_{x\in D_{\text{poison}}}\frac{\omega(x)=y_{\text{target}}}{|D_{\text{poison}}|} \tag{10}$$

4.2　实验结果分析

4.2.1　后门攻击准确性分析

本节评估了在MLP，CNN，AlexNet以及ResNet18模型架构下，针对MNIST，FMNIST，CIFAR10和CIFAR100数据集实施的后门攻击效果，对比了模型在未遭受任何攻击情况下的准确率，以及在遭受像素后门攻击、标签翻转攻击以及KDFLBD等不同类型攻击后的MTA与ASR。

表2完整列出了模型未遭受任何攻击时在MNIST，FMNIST，CIFAR10和CIFAR100数据集上的模型准确率，由于受到数据分布的影响，non-iid场景的准确率整体低于iid场景的准确率。

表2　无攻击情况下各数据集准确率的比较

Table 2　Comparison of accuracy rates for each dataset under no-attack conditions

(%)

数据集	MLP/ResNet18		CNN/AlexNet	
	iid	non-iid	iid	non-iid
MNIST	97.22	96.30	86.37	86.33
FMNIST	86.68	84.91	75.26	72.42
CIFAR10	—	—	77.93	73.48
CIFAR100	43.48	42.74	—	—

表3列出了在训练迭代过程中使用MLP模型在像素后门攻击、标签翻转攻击和KDFLBD攻击方法下的MTA和ASR。在存在攻击的情况下，MTA整体低于无攻击情况。

表3　MLP模型下的不同数据集和攻击方法在iid和non-iid场景下的MTA和ASR

Table 3　Comparison of MTA and ASR for different datasets and attack methods under iid and non-iid scenarios with the MLP model

(%)

数据集	攻击方法	iid		non-iid	
		MTA	*ASR*	*MTA*	*ASR*
MNIST	像素后门	97.29	97.18	96.83	81.32
	标签翻转	97.36	51.56	95.19	46.30
	KDFLBD	97.19	97.47	96.54	97.37
FMNIST	像素后门	85.99	96.30	84.88	70.00
	标签翻转	86.10	43.60	84.50	65.20
	KDFLBD	85.83	96.40	84.91	82.90

表4列出了使用CNN，AlexNet和ResNet18模型在3种攻击方法下的MTA和ASR。KDFLBD能够在MTA与ASR之间达到一个良好的平衡，并且在各种场景下KDFLBD的ASR都高于其他方法，此现象在non-iid场景下更加明显。

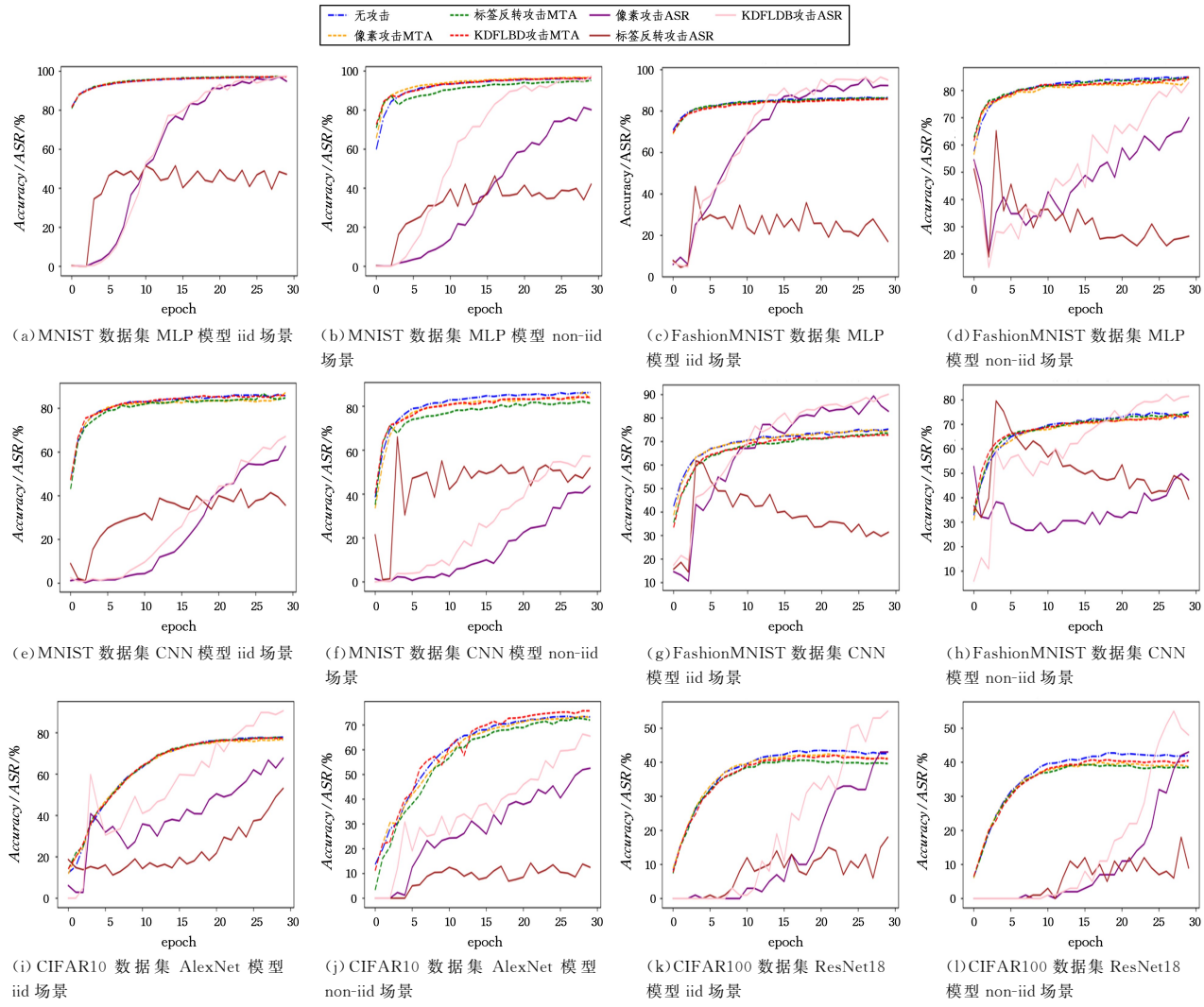
表4　CNN，AlexNet和ResNet18模型下的不同数据集和攻击方法在iid和non-iid场景下的MTA和ASR

Table 4　Comparison of MTA and ASR for different datasets and attack methods under iid and non-iid scenarios with the CNN, AlexNet and ResNet18 model

(%)

数据集	攻击方法	iid		non-iid	
		MTA	*ASR*	*MTA*	*ASR*
MNIST	像素后门	87.19	62.55	86.54	43.58
	标签翻转	86.74	43.09	82.28	51.95
	KDFLBD	86.08	67.12	84.25	57.39
FMNIST	像素后门	74.93	89.40	73.51	49.80
	标签翻转	73.56	61.80	74.00	39.50
	KDFLBD	72.77	90.10	74.03	82.30
CIFAR10	像素后门	76.63	67.70	73.40	52.50
	标签翻转	77.36	53.10	71.99	13.80
	KDFLBD	77.31	90.60	75.69	66.30
CIFAR100 (ResNet18)	像素后门	42.39	43.56	40.02	43.10
	标签翻转	40.59	18.91	39.35	18.95
	KDFLBD	42.12	55.01	40.75	55.03

在训练过程中，KDFLBD的MTA和ASR效果整体优于像素后门攻击与翻转攻击的效果，如图5所示。



(a) MNIST 数据集 MLP 模型 iid 场景 (b) MNIST 数据集 MLP 模型 non-iid 场景 (c) FashionMNIST 数据集 MLP 模型 iid 场景 (d) FashionMNIST 数据集 MLP 模型 non-iid 场景

(e) MNIST 数据集 CNN 模型 iid 场景 (f) MNIST 数据集 CNN 模型 non-iid 场景 (g) FashionMNIST 数据集 CNN 模型 iid 场景 (h) FashionMNIST 数据集 CNN 模型 non-iid 场景

(i) CIFAR10 数据集 AlexNet 模型 iid 场景 (j) CIFAR10 数据集 AlexNet 模型 non-iid 场景 (k) CIFAR100 数据集 ResNet18 模型 iid 场景 (l) CIFAR100 数据集 ResNet18 模型 non-iid 场景

图 5 不同数据集、模型和攻击方法在 iid 和 non-iid 场景下的 MTA 和 ASR 对比

Fig. 5 Comparison of MTA and ASR in iid and non-iid scenarios using different datasets, models, and attack methods

从图 5 中可以看出，随着恶意训练的不断进行，MTA 会趋于拟合，但整体不高于无攻击的场景，KDFLBD 的 MTA 略高于其他两种方法。当第三轮训练攻击者加入时，ASR 呈现集中式膨胀，这种现象在 CNN 和 AlexNet 模型中更加明显，而对于更复杂的模型 ResNet18 来说，攻击效果的体现会滞后多轮。其中，像素攻击的 ASR 在整体上低于 KDFLBD 方法，标签翻转攻击的 ASR 保持在一定范围内震荡，并低于最终拟合的 KDFLBD，3 种方法的 ASR 差距在 non-iid 场景中更加明显。

4.2.2 后门攻击隐蔽性分析

在触发器隐蔽性方面，图 6 展示了实验中经过混合的合成数据集图片，将蒸馏后的数据集和干净样本进行叠加，从而得到浓缩的毒化数据集。可以观察到，合成数据集引入的扰动极为微小，表现出了显著的隐蔽性，人类视觉几乎无法区分。因此服务器端的审查者很难通过参数反推数据集的方法判断恶意客户端。

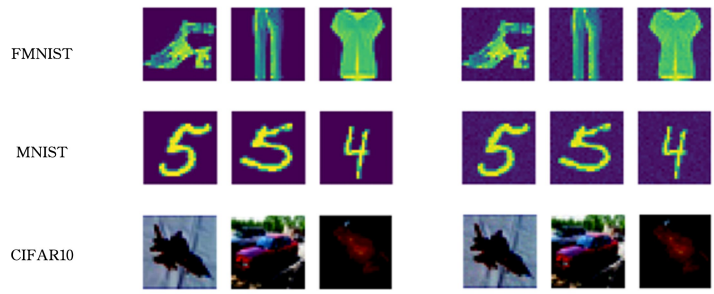


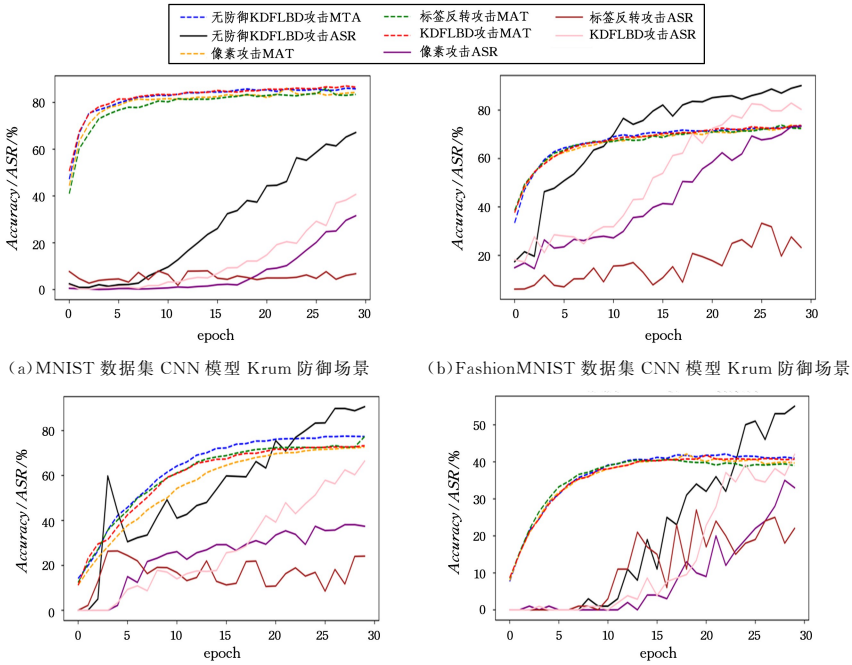
图 6 合成数据集中触发器的隐蔽性

Fig. 6 Concealment of triggers in synthetic dataset

在模型隐蔽性方面，图 7 展示了在服务器使用 Krum 算法进行聚合防御时，使用 CNN，AlexNet 和 ResNet18 模型在 3 种攻击方法下的 MTA 和 ASR。

存在防御情况下的 ASR 整体低于无防御情况下的

ASR，每轮聚合都会将一部分恶意客户端的模型在聚合之前丢弃。相较于前两种攻击方法，KDFLBD 可以在整体上取得更高的 ASR 并通过双分支损失函数策略调整模型梯度下降方向，使其更偏向善意模型，以躲避服务端防御算法的检测。



(a) MNIST 数据集 CNN 模型 Krum 防御场景　(b) FashionMNIST 数据集 CNN 模型 Krum 防御场景

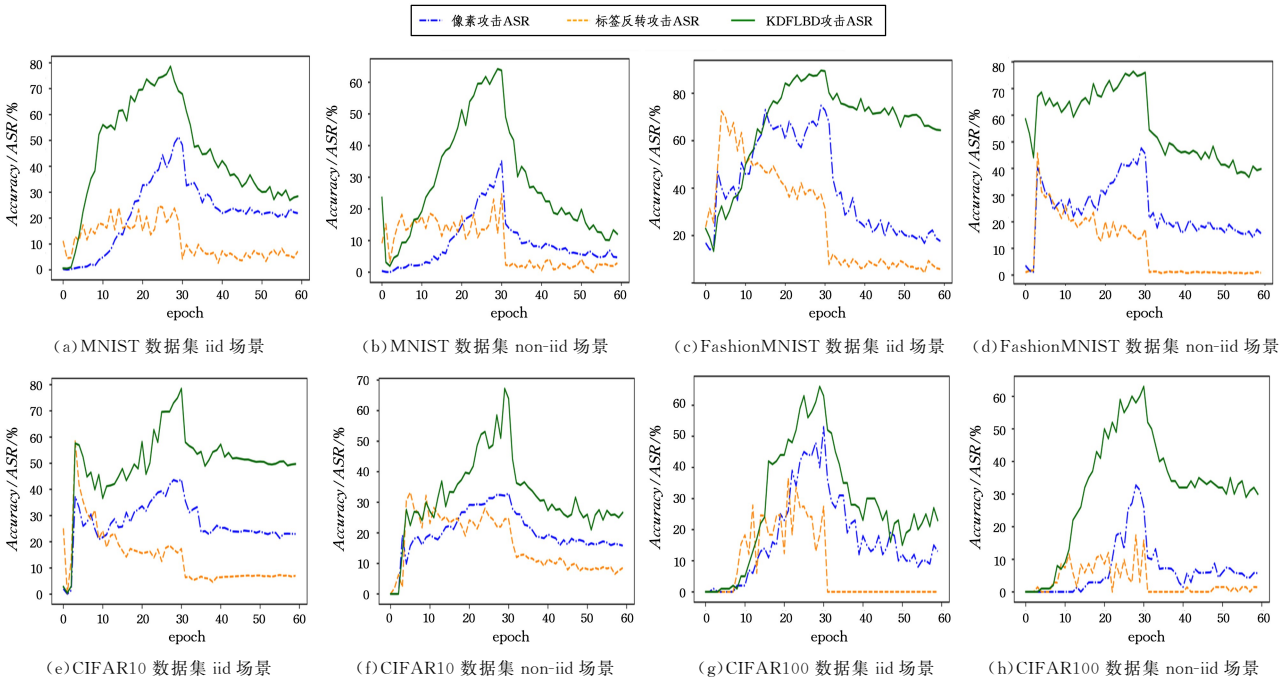
(c) CIFAR10 数据集 AlexNet 模型 Krum 防御场景　(d) CIFAR100 数据集 ResNet18 模型 Krum 防御场景

图 7　不同数据集、模型和攻击方法在 Krum 防御下的对比

Fig. 7　Comparison of different datasets, models and attack methods under Krum defense

4.2.3　后门攻击持久性分析

在全局模型完成 30 轮的训练后，恶意客户端停止参与，再继续训练 30 轮。通过对比不同数据集在不同数据划分方式下 3 种攻击的 ASR 变化趋势可以看出，ASR 总体呈下降趋势，其中 KDFLBD 的 ASR 整体高于像素后门攻击和标签翻转攻击的 ASR。具体结果如图 8 所示。



(a) MNIST 数据集 iid 场景　(b) MNIST 数据集 non-iid 场景　(c) FashionMNIST 数据集 iid 场景　(d) FashionMNIST 数据集 non-iid 场景

(e) CIFAR10 数据集 iid 场景　(f) CIFAR10 数据集 non-iid 场景　(g) CIFAR100 数据集 iid 场景　(h) CIFAR100 数据集 non-iid 场景

图 8　不同数据集和攻击方法在 ASR 持久性方面的对比

Fig. 8　Comparison of ASR persistence using different datasets and attack methods

从图 8 中可以看出，随着恶意客户端的退出，各个场景下的 ASR 会迅速下降，其中标签翻转攻击受到的影响高于其他 3 种方法，KDFLBD 在下降过程中整体高于像素攻击，3 种方法的 ASR 差距在 non-iid 场景中更加明显。

4.2.4　消融实验

为了深入理解蒸馏温度超参数 T 以及损失混合比例 σ 对模型性能的影响，对 KDFLBD 开展消融实验，通过对超参数进行不同取值来分析它们对攻击性能的影响。

温度超参数 T 用于生成软概率分布以及各样本属于每个类别的概率，从而导致在类别输出上产生细微变化，这些变化直接影响了模型之间学习的知识细致程度。将 T 分别取 1,3,6,10 来训练学生模型，表 5 列出了不同 T 值下的模型准确度和后门攻击成功率。随着温度的增加，模型输出的概率发布变得平滑，能够更加有效地提取各个类别间的知识，但过高的温度可能会破坏知识结构，丢失关键的区分信息，导致 MTA 下降。

表 5 不同蒸馏温度 T 下的 MTA 和 ASR

Table 5 Comparison of MTA and ASR at different distillation temperatures T

(%)

数据集	评估指标	$T=1$	$T=3$	$T=6$	$T=10$
MNIST	*MTA*	86.06	86.08	86.77	85.57
	ASR	66.25	67.12	56.81	43.09
FMNIST	*MTA*	72.24	72.77	72.33	72.19
	ASR	79.90	90.10	79.60	78.00
CIFAR10	*MTA*	77.13	77.31	77.42	77.06
	ASR	87.40	90.60	83.80	78.70
CIFAR100	*MTA*	41.37	42.12	41.75	40.90
	ASR	57.05	55.01	53.06	52.89

更低的温度使得模型更加关注数据的细节部分，并对噪声更加敏感，放大了噪声的负面影响，导致 ASR 上升。

下面对损失混合比例 σ 进行评估，其控制着学生模型的损失函数和梯度下降的方向。表 6 列出了不同混合比例 σ 下的模型准确度和后门攻击成功率。

表 6 不同损失混合比例 σ 下的 MTA 和 ASR

Table 6 Comparison of MTA and ASR with different loss mixing ratios σ

(%)

数据集	评估指标	$\sigma=0.1$	$\sigma=0.3$	$\sigma=0.5$	$\sigma=0.9$
MNIST	*MTA*	86.29	86.35	86.08	86.27
	ASR	62.94	63.48	67.12	67.19
FMNIST	*MTA*	73.22	72.32	72.77	72.00
	ASR	83.43	84.40	90.10	84.71
CIFAR10	*MTA*	81.30	77.21	77.31	77.88
	ASR	76.93	82.60	90.60	86.30
CIFAR100	*MTA*	42.28	41.46	42.12	40.86
	ASR	52.00	53.19	55.01	61.00

当 $\sigma=0.9$ 时，ASR 的值普遍高于其他情况，σ 越大代表学生模型越倾向于攻击者模型。当 $\sigma=0.1$ 时，代表学生模型越倾向于纯净模型，从而获取更高的 MTA。

4.2.5 防御框架

针对 KDFLBD 的防御机制，可构建细粒度净化框架。在服务器了解此类攻击算法流程后，可以借助其他计算资源复现恶意客户端的攻击过程，建立攻击特征库 $A=\{R(x,y), D_T, \widetilde{D}, A_{\text{poison}}, A_{\text{clear}}\}$，将此部分独立完成可降低对当前服务器资源的需求。服务器利用保留的少量验证集 V 复现后门嵌入算法(算法 2)得到恶意数据集 $\widetilde{D}$，然后根据神经元替换算法(算法 4)得到激活值(A_{poison}，A_{clear})和 Z 分数，并按照 Z 分数选取超过一定阈值的神经元作为可能污染的神经元集群。

基于攻击特征库的定位结果，对每个客户端上传的模型实施细粒度神经元替换策略。将部分验证集 V 训练后的净化神经元 $\omega'[i]$ 按位置映射关系进行替换，以此来缓解恶意客户端对模型的毒化攻击，同时维持联邦学习分布式的特性。净化过程仅涉及全连接层的局部微调，对服务器的资源需求较小。除此之外，基于神经元位置的替换策略支持多种深度网络架构的适配。

结束语 本研究提出了一种基于知识蒸馏的联邦学习后门生成方案，其能够在不影响主任务准确率的情况下提高后门攻击的成功率。该方案通过蒸馏恶意数据集得到浓缩的毒化数据集，并将其用于教师模型的训练，然后将教师模型的暗知识传递给学生模型以获取浓缩的恶意神经元，最后按照 Z 分数排序结果替换原始模型的神经元，增强后门攻击的隐蔽性。

未来计划继续探索通过蒸馏注入的触发器与后门模型之间的关系，进一步验证攻击方法在其他模型上的泛用性，并实现高效的防御机制，检测并消除通过知识蒸馏注入的触发器，从而降低联邦学习中后门攻击对模型的恶意影响。

参 考 文 献

[1] MOORE I N, SNYDER S L, MILLER C, et al. Confidentiality and Privacy in Health Care from the Patient's Perspective: Does HIPPA Help? [J]. Health Matrix, 2007, 17: 215.

[2] VOIGT P, VON DEM BUSSCHE A. The eu general data protection regulation (gdpr): A Practical Guide (1st Ed.) [M]. Cham: Springer International Publishing, 2017.

[3] CHENG X. On the personal information processing rules in our country's personal information protection law [J]. Tsinghua Law, 2021, 15(3): 55-73.

[4] 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.

[5] 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.

[6] XUE M, NI S, WU Y, et al. Imperceptible and multi-channel backdoor attack [J]. Applied Intelligence, 2024, 54(1): 1099-1116.

[7] BAGDASARYAN E, SHMATIKOV V. Blind backdoors in deep learning models [C]// 30th USENIX Security Symposium (USENIX Security 21). 2021: 1505-1521.

[8] RAWAT A, LEVACHER K, SINN M. The devil is in the GAN: backdoor attacks and defenses in deep generative models [C]// European Symposium on Research in Computer Security. Cham: Springer Nature Switzerland, 2022: 776-783.

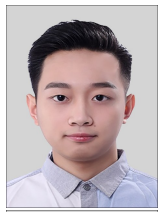
[9] NGUYEN T D, RIEGER P, MIETTINEN M, et al. Poisoning attacks on federated learning-based IoT intrusion detection system [C] // Proc. Workshop Decentralized IoT Syst. Secur. (DISS). 2020: 1-7.

[10] LIU Y, GARG S, NIE J, et al. Deep anomaly detection for time-series data in industrial iot: A communication-efficient on-device federated learning approach [J]. IEEE Internet of Things Journal, 2021(8): 6348-6358.

[11] CHEN M, SURESH A T, MATHEWS R, et al. Federated learning of n-gram language models [J]. arXiv: 1910. 03432, 2019.

[12] LI T, SAHU A K, ZAHEER M, et al. Federated optimization in heterogeneous networks [C]// Proceedings of Machine Learning

and Systems. 2020：429-450.
[13] LI X，JIANG M，ZHANG X，et al. Fedbn：Federated learning on non-iid features via local batch normalization[J]. arXiv：2102.07623，2021.
[14] LI Q，HE B，SONG D. Model-contrastive federated learning [C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021：10713-10722.
[15] GU T，DOLAN-GAVITT B，GARG S. Badnets：Identifying vulnerabilities in the machine learning model supply chain[J]. arXiv：1708.06733，2017.
[16] ALBERTI M，PONDENKANDATH V，WURSCH M，et al. Are you tampering with my data? [C]// Proceedings of the European Conference on Computer Vision(ECCV). 2018.
[17] BARNI M，KALLAS K，TONDI B. A new backdoor attack in cnns by training set corruption without label poisoning[C]// 2019 IEEE International Conference on Image Processing (ICIP). IEEE，2019：101-105.
[18] XIAO Q，CHEN Y，SHEN C，et al. Seeing is not believing：Camouflage attacks on image scaling algorithms[C]// 28th USENIX Security Symposium(USENIX Security 19). 2019：443-460.
[19] LI Y，LI Y，WU B，et al. Invisible backdoor attack with sample-specific triggers[C]// Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021：16463-16472.
[20] SHAFAHI A，HUANG W R，NAJIBI M，et al. Poison frogs! targeted clean-label poisoning attacks on neural networks[J]. arXiv：1804.00792，2018.
[21] GAO Y，LI Y，ZHU L，et al. Not all samples are born equal：Towards effective clean-label backdoor attacks[J]. Pattern Recognition，2023，139：109512.
[22] LIN J，XU L，LIU Y，et al. Composite backdoor attack for deep neural network by mixing existing benign features[C]// Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security. 2020：113-131.
[23] WANG H，SREENIVASAN K，RAJPUT S，et al. Attack of the tails：Yes，you really can backdoor federated learning[J]. Advances in Neural Information Processing Systems，2020，33：16070-16084.
[24] YOO K Y，KWAK N. Backdoor attacks in federated learning by rare embeddings and gradient ensembling [J]. arXiv：2204.14017，2022.
[25] ZHANG J，CHEN B，CHENG X，et al. PoisonGAN：Generative poisoning attacks against federated learning in edge computing systems[J]. IEEE Internet of Things Journal，2020，8(5)：3310-3322.
[26] GONG X，CHEN Y，HUANG H，et al. Coordinated backdoor attacks against federated learning with model-dependent triggers [J]. IEEE Network，2022，36(1)：84-90.
[27] XIE C，HUANG K，CHEN P Y，et al. Dba：Distributed backdoor attacks against federated learning[C]// International Conference on Learning Representations. 2019.
[28] SUN Z，KAIROUZ P，SURESH A T，et al. Can you really backdoor federated learning? [J]. arXiv：1911.07963，2019.
[29] LIU Y，YI Z，CHEN T. Backdoor attacks and defenses in feature-partitioned collaborative learning[J]. arXiv：2007.03608，2020.
[30] ZHOU X，XU M，WU Y，et al. Deep model poisoning attack on federated learning[J]. Future Internet，2021，13(3)：73.
[31] ZHANG Z，PANDA A，SONG L，et al. Neurotoxin：Durable backdoors in federated learning[C]// International Conference on Machine Learning. PMLR，2022：26429-26446.
[32] BUCILUǍ C，CARUANA R，NICULESCU-MIZIL A. Model compression[C]// Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2006：535-541.
[33] HINTON G，VINYALS O，DEAN J. Distilling the knowledge in a neural network[J]. arXiv：1503.02531，2015.
[34] CAZENAVETTE G，WANG T，TORRALBA A，et al. Dataset distillation by matching training trajectories[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022：4750-4759.
[35] NGUYEN T，CHEN Z，LEE J. Dataset meta-learning from kernel ridge-regression[J]. arXiv：2011.00050，2020.
[36] NGUYEN T，NOVAK R，XIAO L，et al. Dataset distillation with infinitely wide convolutional networks[J]. Advances in Neural Information Processing Systems，2021，34：5186-5198.
[37] ZHAO B，BILEN H. Dataset condensation with differentiable siamese augmentation[C]// International Conference on Machine Learning. PMLR，2021：12674-12685.
[38] ZHAO B，MOPURI K R，BILEN H. Datasetcondensation with gradient matching[J]. arXiv：2006.05929，2020.
[39] WANG T，ZHU J Y，TORRALBA A，et al. Dataset distillation [J]. arXiv：1811.10959，2018.
[40] RUBINSTEIN R. The cross-entropy method for combinatorial and continuous optimization[J]. Methodology and Computing in Applied Probability，1999，1(2)：127-190.
[41] LECUN Y，BOTTOU L，BENGIO Y，et al. Gradient-based learning applied to document recognition[C]// Proceedings of the IEEE. 2002：2278-2324.
[42] XIAO H，RASUL K，VOLLGRAF R. Fashion-mnist：a novel image dataset for benchmarking machine learning algorithms [J]. arXiv：1708.07747，2017.
[43] KRIZHEVSKY A，HINTON G. Learning multiple layers of features from tiny images[J/OL]. https：//www.cs.utoronto.ca/~kriz/learning-features-2009-TR.pdf.
[44] CAO X，JIA J，GONG N Z. Provably secure federated learning against malicious clients[C]// Proceedings of the AAAI Conference on Artificial Intelligence. 2021：6885-6893.
[45] KRIZHEVSKY A，SUTSKEVER I，HINTON G E. Imagenet classification with deep convolutional neural networks[J/OL]. https：//proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.
[46] NGUYEN T D，NGUYEN T，LE NGUYEN P，et al. Backdoor attacks and defenses in federated learning：Survey，challenges and future research directions[J]. Engineering Applications of Artificial Intelligence，2024，127：107166.



ZHAO Tong，born in 1998，postgraduate，is a member of CCF (No. W0214G). His main research interests include data security and federated learning.

CHEN Xuebin，born in 1970，Ph.D，professor，is a outstanding member of CCF (No. 13654D). His main research interests include big data security，Internet of security and network security.

（责任编辑：柯颖）