软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2023,34(6):2833−2864 [doi: 10.13328/j.cnki.jos.006658]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

联邦学习模型安全与隐私研究进展*

顾育豪, 白跃彬

(北京航空航天大学 计算机学院, 北京 100191)

通信作者: 白跃彬, E-mail: byb@buaa.edu.cn

摘　要: 随着数据孤岛现象的出现和个人隐私保护的重视, 集中学习的应用模式受到制约, 而联邦学习作为一个分布式机器学习框架, 可以在不泄露用户数据的前提下完成模型训练, 从诞生之初就备受关注. 伴随着联邦学习应用的推广, 其安全性和隐私保护能力也开始受到质疑. 对近年来国内外学者在联邦学习模型安全与隐私的研究成果进行了系统总结与分析. 首先, 介绍联邦学习的背景知识, 明确其定义和工作流程, 并分析存在的脆弱点. 其次, 分别对联邦学习存在的安全威胁和隐私风险进行系统分析和对比, 并归纳总结现有的防护手段. 最后, 展望未来的研究挑战和方向.

关键词: 联邦学习; 安全和隐私; 投毒攻击; 推断攻击; 防护方法

中图法分类号: TP309

中文引用格式: 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, 2023, 34(6): 2833–2864. http://www.jos.org.cn/1000-9825/6658.htm

英文引用格式: Gu YH, Bai YB. Survey on Security and Privacy of Federated Learning Models. Ruan Jian Xue Bao/Journal of Software, 2023, 34(6): 2833–2864 (in Chinese). http://www.jos.org.cn/1000-9825/6658.htm

Survey on Security and Privacy of Federated Learning Models

GU Yu-Hao, BAI Yue-Bin

(School of Computer Science and Engineering, Beihang University, Beijing 100191, China)

Abstract: As data silos emerge and importance is attached to personal privacy protection, the application modes of centralized learning are restricted, whereas federated learning has attracted great attention since it appeared owing to the fact that it, as a distributed machine learning framework, can accomplish model training without leaking users' data. As federated learning is increasingly widely applied, its security and privacy protection capability have also begun to be questioned. This study offers a systematic summary and analysis of the research achievements domestic and foreign researchers have made in recent years in the security and privacy of federated learning models. Specifically, this study outlines the background of federated learning, clarifies its definition and workflow, and analyzes its vulnerabilities. Then, the security threats and privacy risks against federated learning are systematically analyzed and compared respectively, and the existing defense methods are summarized. Finally, the prospects of this research area and the challenges ahead are presented.

Key words: federated learning; security and privacy; poisoning attack; inference attack; defense method

近年来, 机器学习 (machine learning) 技术蓬勃发展, 在社会工作生活各个领域中得到广泛应用, 如人脸识别、智慧医疗和自动驾驶等, 并取得巨大的成功. 机器学习的目标是从大量数据中学习到一个模型, 训练后的模型可以对新的未知数据预测结果, 因此模型的性能与训练数据的数量和质量密切相关. 传统的机器学习应用基本都采取集中学习[1]的模式, 即由服务提供商集中收集用户数据, 在服务器或数据中心训练好模型后, 将模型开放给用户使用. 但是, 目前存在两大要素制约了集中学习的进一步推广.

* 基金项目: 国家自然科学基金 (61732002, 61572062)
收稿时间: 2021-04-08; 修改时间: 2022-01-02; 采用时间: 2022-02-16; jos 在线出版时间: 2022-09-20
CNKI 网络首发时间: 2022-12-10

(1) 数据孤岛

随着信息化、智能化进程的发展, 各个企业或同一企业的各个部门都存储了大量的应用数据, 但是数据的定义和组织方式都不尽相同, 形成一座座相互独立且无法关联的“孤岛”, 影响数据的流通和应用. 数据集成整合的难度和成本严重限制了集中学习的推广应用.

(2) 个人隐私保护的重视

近年来, 个人数据泄露的事件层出不穷, 如 2018 年 Facebook 数据泄露事件等. 这些事件引起了国家和公众对于个人隐私保护的关注. 各个国家都开始出台数据隐私保护相关的法律法规, 如欧盟 2018 年 5 月 25 日出台的《通用数据保护条例》(general data protection regulation, GDPR)[2], 以及中国 2017 年实施的《中华人民共和国网络安全法》等. 这些法律法规要求公司企业必须在用户同意的前提下才可以收集个人数据, 且需要防止用户数据泄露. 此外, 个人隐私保护意识的兴起也导致用户不愿轻易共享自己的隐私数据. 严格的法律法规和个人隐私保护意识导致训练数据的收集愈发困难, 为集中学习提出了巨大的挑战.

为应对上述两个问题, 联邦学习 (federated learning) 应运而生. 联邦学习, 又名联盟学习或联合学习, 是一种由多个客户端和一个聚合服务器参与的分布式机器学习架构. 客户端既可以是个人的终端设备 (如手机等), 也可以代表不同的部门或企业, 它负责保存用户的个人数据或组织的私有数据. 客户端在本地训练模型, 并将训练后的模型参数发送给聚合服务器. 聚合服务器负责聚合部分或所有客户端的模型参数, 将聚合后的模型同步到客户端开始新一轮的训练. 这种联合协作训练的方式可以在保证模型性能的前提下, 避免个人数据的泄露, 并有效解决数据孤岛的问题.

联邦学习自 2016 年谷歌[3]提出后便引起学术界和工业界的强烈关注, 并涌现出许多实际应用, 如谷歌最初将其应用在安卓手机上的 Gboard APP (the Google keyboard, 谷歌键盘输入系统), 用于预测用户后续要输入的内容 (如图 1 所示)[4]. 用户手机从服务器下载预测模型, 基于本地用户数据进行训练微调, 并上传微调后的模型参数, 不断优化服务器的全局模型. 此外, 联邦学习也被广泛应用于工业[5,6]、医疗[7–11]和物联网[12]等领域.

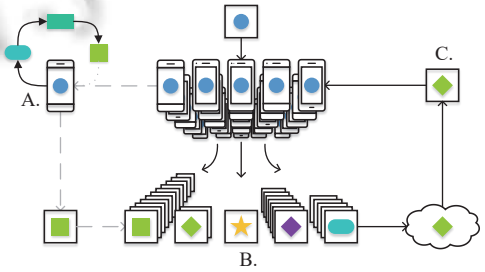

图 1 基于联邦学习的语言预测应用 Gboard

随着联邦学习的发展应用, 其安全性与隐私性逐渐引起学术界的关注. 与集中学习相比, 联邦学习的模型参数共享和多方通信协作机制引入了新的攻击面. 近年来, 许多学者对联邦学习的安全威胁进行深入研究, 提出一系列攻击手段和防护方案. 除安全性外, 学者也发现联邦学习存在诸如成员推断攻击等隐性泄露的风险. 这些将严重影响联邦学习的实际部署应用, 因此本文对目前联邦学习模型的安全与隐私研究工作进行系统的整理和科学的归纳总结, 分析联邦学习面临的安全隐私风险及挑战, 为后续学者进行相关研究时提供指导.

本文第 1 节主要介绍联邦学习的背景知识, 明确其定义和工作流程, 并分析其存在的脆弱点. 第 2 节对联邦学习存在的安全威胁进行系统的整理和分析, 归纳现有的防护方法, 并对集中学习和联邦学习在安全问题上的共性与差异进行分析. 第 3 节总结联邦学习的隐私风险以及隐私保护方面的研究进展, 讨论集中学习和联邦学习在隐私风险的差异. 第 4 节展望未来的研究方向, 提出联邦学习安全和隐私领域亟待解决的重要问题. 第 5 节总结全文.

1 背景知识

1.1 定 义

联邦学习是一种分布式的机器学习框架, 最早是由谷歌的 McMahan 等人提出并落地应用[3]. 他们在不泄露用

户个人数据的前提下, 利用分布在不同手机的数据集训练统一的机器学习模型. 微众银行 Yang 等人对谷歌提出的联邦学习概念进行扩展, 将其推广成所有隐私保护的协作机器学习技术的一般概念, 以涵盖组织间不同的协作学习场景[13]. 本文采用微众银行团队扩展后的联邦学习定义, 具体如下[13].

假定 n 位参与方$\{P_1, P_2, \ldots, P_n\}$协作, 通过使用各自的数据集$\{D_1, D_2, \ldots, D_n\}$训练机器学习模型. 每个数据集包含多个数据样本和数据特征 (feature), 部分数据集可能还包含标签信息 (label). 设数据集的数据特征空间为 F, 标签空间为 L, 样本 ID 空间为 I, 三者构成完整的训练数据集, 即:

$$D_i = (I_i, L_i, F_i), \forall i \in [1, n] \tag{1}$$

集中学习是将所有参与方的数据集合存储形成数据集 $D = D_1 \cup D_2 \cup \ldots \cup D_n$, 基于 D 训练模型 M_{sum}. 而联邦学习是一个由参与方联合训练模型 M_{fed} 的框架, 可以保证在训练过程中, P_i 不会向其他参与方公开其拥有的数据集 D_i. 设 V_{sum} 和 V_{fed} 分别表示模型 M_{sum} 和 M_{fed} 的性能度量值 (如准确率、$F1$ 分数等), δ 为一个非负实数, 当满足下列条件时, 称联邦学习模型具有 δ 的性能损失:

$$|V_{\text{fed}} - V_{\text{sum}}| < \delta \tag{2}$$

联邦学习以少量性能损失换取额外的隐私保护和数据安全. 为保证联邦学习模型的有效性, δ 的值在实际应用中应尽可能接近 0.

根据不同参与方的数据集在特征空间 F、标签空间 L 和样本 ID 空间 I 的分布情况, 可以将联邦学习分为以下 3 类[13].

(1) 横向联邦学习

横向联邦学习是针对多个参与方的数据集拥有相同的数据特征, 但样本不同的场景, 其定义如下:

$$F_i = F_j, L_i = L_j, I_i \neq I_j, \forall i, j \in [1, n] \text{且} i \neq j \tag{3}$$

谷歌的 Gboard 是典型的横向联邦学习应用[4]. 横向联邦学习还可以用于不同医院间的疾病诊断模型, 以及物联网设备间的协调合作. 横向联邦学习可以有效扩大训练样本的数量, 是目前最常见的联邦学习类型.

(2) 纵向联邦学习

纵向联邦学习适用于多个参与方的数据集具有相同的样本 ID 空间, 但特征空间不同的场景, 其定义如下:

$$F_i \neq F_j, L_i \neq L_j, I_i = I_j, \forall i, j \in [1, n] \text{且} i \neq j \tag{4}$$

例如某个地区的银行和电子商务公司拥有的数据集都包含本地区的居民, 样本 ID 空间有大量交叉, 但数据特征却完全不同. 其中银行的数据是描述用户的收支行为和资金状况, 而电子商务公司保存的是用户对各种商品的浏览与购买记录. 两个公司可以利用纵向联邦学习联合训练一个用户购买商品的预测模型.

(3) 联邦迁移学习

联邦迁移学习是针对两个参与方的数据集特征不同且样本也不同的应用场景, 其定义如下:

$$F_i \neq F_j, L_i \neq L_j, I_i \neq I_j, \forall i, j \in [1, n] \text{且} i \neq j \tag{5}$$

例如不同地区的银行和电子商务公司的数据集样本空间中只有少量重叠. 他们可以利用联邦迁移学习进行合作, 基于有限的公共样本集学习两个特征空间的公共表示.

目前针对联邦学习模型的安全与隐私研究主要集中在横向联邦学习, 因此下文如无特殊说明, 联邦学习均指代横向联邦学习.

1.2 系统架构

图 2 为联邦学习系统的典型架构, 架构中包含两类角色: 多个参与方 (也称客户或用户) 和一个聚合服务器. 每个参与方拥有完整数据特征的数据集, 且数据样本之间没有交集或交集很小. 它们可以联合起来训练一个统一且性能更好的全局模型, 具体的训练过程如下.

(1) 模型初始化: 聚合服务器选定目标模型的结构和超参数, 并初始化模型的权重 (基于自身拥有的数据 D_{server} 进行训练或随机初始化), 生成初始的全局模型.

(2) 模型广播: 通过聚合服务器广播或参与方主动下载的方式, 聚合服务器将当前全局模型的权重共享给所有参与方.

(3) 参与方训练: 参与方基于共享的全局模型, 利用本地保存的私有数据训练微调本地模型, 并计算本地模型的权重更新.

(4) 模型聚合: 聚合服务器从参与方收集模型的权重更新, 根据业务需求采用不同的算法进行聚合. 常见的聚合算法包括 FedAvg[3]、Krum[14]、Trimmed-mean[15]和 Median[15]等. 在这过程中为了提高效率, 聚合服务器可以选择只收集部分参与方的模型更新进行聚合.

(5) 更新全局模型: 聚合服务器基于计算的聚合结果更新全局模型的参数.

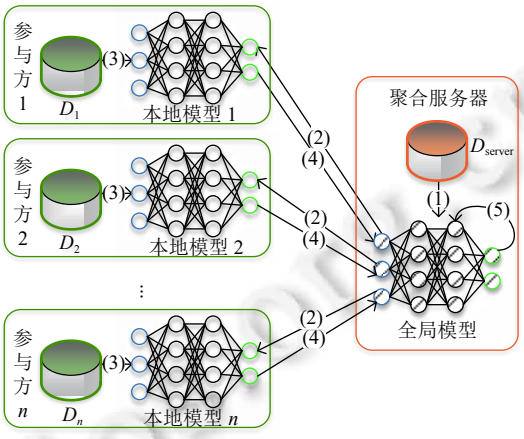

图 2 联邦学习系统的架构

图 2 中步骤 (2)–(5) 将会持续迭代进行, 直至全局模型收敛或者达到最大迭代次数或超过最长训练时间.

1.3 脆弱点分析

在总结联邦学习模型的安全与隐私研究之前, 本文首先对联邦学习系统的脆弱点进行分析. 如图 2 的架构所示, 联邦学习需要参与方和聚合服务器之间通信协作, 因此存在以下脆弱点.

(1) 通信协议

在训练模型的迭代过程中, 参与方需要和聚合服务器进行数据通信. 参与方需要将本地的模型更新发送给聚合服务器, 而聚合服务器也需要下发新的全局模型. 更新中包含模型的梯度信息, 可用于推断参与方的训练数据, 泄露参与方的隐私[16]. 因此, 通信协议的可靠性和保密性决定了联邦学习系统的安全性.

(2) 聚合服务器

聚合服务器负责初始化模型参数、聚合参与方的模型更新和下发全局模型. 若服务器被攻陷, 攻击者可以随意发布恶意模型, 影响参与方的本地应用. 另外, 服务器可以查看各个参与方发送的模型更新, 诚实但好奇 (honest but curious) 的服务器可以基于模型更新重构参与方的本地数据[17].

(3) 参与方

参与方可以通过上传恶意的模型更新破坏聚合后的全局模型. 目前常见的联邦学习应用的参与方都是个人用户 (如 Gboard 应用[4]). 与聚合服务器相比, 个人用户的安全防护措施薄弱, 攻击成本低, 攻击者可以通过入侵普通用户或者注册新用户等手段, 轻易加入到联邦学习的训练过程中, 通过伪造本地数据、修改模型更新等方法攻击全局模型, 还可以勾结多个恶意方同时发动攻击, 增强攻击效果. 因此, 参与方是联邦学习系统中最大的脆弱点[18].

2 安全威胁与防护

在集中学习的发展过程中, 许多学者对其安全性进行深入研究, 发现其中存在的安全威胁, 如训练阶段的投毒攻击 (poisoning attack)[19]和推理阶段的对抗样本攻击 (adversarial examples attack) 等[20]. 联邦学习的推理阶段与集中学习一致, 因此也会面临对抗样本攻击. 而在训练阶段, 联邦学习采用分布式计算的方法, 为整个系统的安全性

研究引入了新的问题与挑战. 本文主要总结面向联邦学习的安全威胁与防护方法, 与集中学习相关的安全研究不在本文的讨论范围内.

本文以联邦学习面临的安全攻击的发生逻辑和顺序对目前主要研究的攻击手段进行分类 (如图 3 所示), 具体可分为数据投毒攻击[21−23]、模型投毒攻击[24−29]、后门攻击[25,30−34]和恶意服务器. 注意, 图 3 的推理阶段在实际应用中还存在对抗样本等攻击手段, 这部分不在本文的讨论范围内.

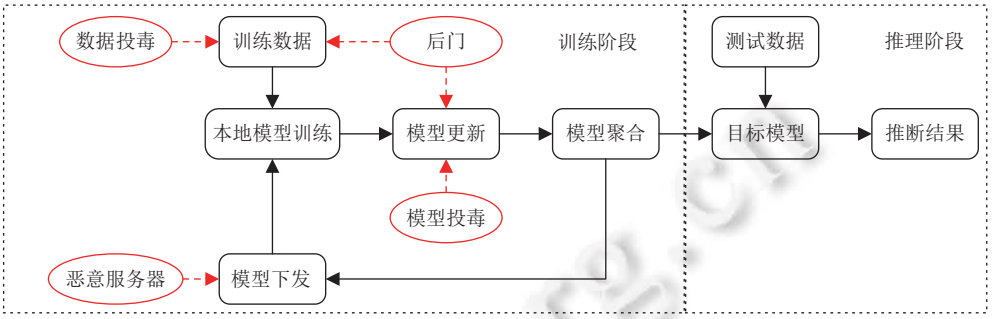

图 3 联邦学习面临的安全攻击

2.1 威胁模型分析

攻击者对联邦学习系统发动不同攻击时有不同的攻击目标, 同时也需要不同的背景知识和能力, 因此本文从攻击者目标、攻击者能力以及攻击者知识 3 个维度对安全攻击的威胁模型 (threat model) 进行分析[29].

(1) 攻击者目标

攻击者的目标是降低联邦学习全局模型的性能 (如准确率、$F1$ 分数等), 根据其具体目标可细分为两类: 非定向 (untargeted) 攻击和定向 (targeted) 攻击. 其中非定向攻击是影响模型对任意输入数据的推理, 而定向攻击只降低模型对特定标签的输入数据的推理准确率, 而不影响或轻度影响其他标签数据的性能. 以自动驾驶应用的交通标志识别模型为例, 非定向攻击是使模型无法识别所有交通标志, 而定向攻击可以使模型将停车标志识别为限速标志, 而不影响其他标志的识别.

(2) 攻击者能力

攻击者能力是指攻击者对联邦学习系统的角色和数据所拥有的操作权限. 在现有的安全研究工作中, 攻击者能力从高到低依次包括: 控制服务器、控制多个参与方、控制单个参与方和控制参与方训练数据. 其中控制服务器和控制参与方是指攻击者可以随意访问修改服务器或参与方的模型和数据, 干扰其执行的操作, 而控制训练数据是指攻击者可以读取、插入或修改参与方的训练数据集. 攻击要求的能力越低, 在实际应用中越容易实施.

(3) 攻击者知识

攻击者知识是指攻击者对目标联邦学习系统的背景知识, 具体包括: 服务器采用的聚合算法、每轮迭代中所有参与方上传的模型更新、参与方训练数据集的数据分布等. 攻击所需知识越少, 在实际应用中越容易实施.

2.2 安全攻击手段

2.2.1 数据投毒

数据投毒攻击 (data poisoning attack) 最早由 Biggio 等人提出[35], 攻击者通过污染训练数据集 (如添加伪造数据或修改已有数据等), 使模型在训练过程中学习错误的对应关系, 从而降低模型的准确性. 在联邦学习系统中, 攻击者可以通过控制参与方或者修改参与方训练数据集等方式, 实施数据投毒攻击. 然而, 聚合算法会削弱数据投毒对全局模型的影响.

标签翻转 (label flipping) 是一种典型的数据投毒攻击, 通过直接修改目标类别的训练数据的标签信息, 使模型将目标标签的特征对应到错误标签, 从而影响模型的推理效果. Tolpegin 等人[21]对联邦学习的标签翻转攻击进行详细的实验分析. 他们从 3 个维度 (恶意方的比例、发动攻击的迭代轮数、每轮迭代中恶意方参与聚合的概率)

对联邦学习的标签翻转攻击效果进行验证, 证明数据投毒攻击会降低联邦学习的安全性, 且恶意方比例的增加会增大对全局模型的负面影响, 并发现可以通过提高恶意方在后期的迭代轮数中参与聚合的概率进一步增强攻击效果. 对于恶意方的训练集中没有目标标签数据的场景, Zhang 等人[22]提出基于生成对抗网络 (generative adversarial nets, GAN)[36]的数据投毒攻击. 恶意方通过在本地部署 GAN, 将每轮聚合后的全局模型作为 GAN 的判别网络 D, 利用 GAN 的生成网络 G 生成模仿目标标签数据的样本, 之后基于生成的数据样本实施标签翻转攻击. 为进一步提升攻击效果, Zhang 等人[23]在原先工作的基础上提出 PoisonGAN 攻击, 通过修改恶意方本地训练的超参数, 在恶意方的模型更新上添加比例系数, 从而提高恶意更新对全局模型的影响力, 扩大生成数据的毒害效果.

在联邦学习中, 因为控制参与方的攻击成本较低, 攻击者可以发动攻击效果更好且更灵活的模型投毒攻击, 所以现有关于数据投毒的研究较少. 但数据投毒攻击在实际应用中对攻击者能力和知识要求最少, 只需要攻击者可以控制参与方的训练数据, 因此具有广泛的实施场景.

2.2.2 模型投毒

模型投毒攻击 (model poisoning attack) 是通过直接修改模型的权重参数对模型进行攻击, 当模型采用随机梯度下降 (stochastic gradient descent) 算法时, 则是修改模型梯度. 在联邦学习的工作流程中, 参与方需要向服务器发送本地的模型更新. 因为参与方的数据和训练过程都是在本地完成, 对服务器不可见, 所以服务器无法对参与方上传的模型更新的真实性进行验证. 这些为攻击者实施模型投毒攻击创造了条件. 恶意方可以构造任意模型更新发送给服务器, 破坏聚合后的全局模型.

联邦学习最常用的聚合算法 FedAvg 是在前一轮全局模型上添加本地模型更新的平均值[3], 后续也出现了其他变种算法, 如 Li 等人[37]提出的 FedProx 算法等. 这些基于线性组合的算法使恶意方可以随意操纵全局模型[14]. 因此学者提出了一系列拜占庭容错的聚合算法, 可以容忍联邦学习网络中出现部分拜占庭节点 (即恶意方), 如 Krum[14]、Trimmed-mean[15]等. 针对这些拜占庭容错算法, 其他学者则开始研究如何利用聚合算法存在的缺陷, 设计可以绕过聚合算法的模型投毒攻击.

针对 Krum[14]等基于 l_p 范数 (即计算比较恶意梯度与正常梯度在 l_p 空间的距离) 的聚合算法, 利用在高维梯度中 l_p 距离度量$||X-Y||_p$ 无法解释 X 和 Y 的差距是源自多维的少量差距叠加还是单一维度的特点, Mhamdi 等人[24]通过在正常梯度的单维参数上添加偏差形成恶意梯度, 并勾结多个恶意方放大该维度的偏差, 从而降低全局模型的准确性. 而 Baruch 等人[25]是利用 Krum[14]和 Trimmed-mean[24]等聚合算法对于恶意梯度远离梯度均值的错误假定, 在多轮迭代中为恶意梯度的每一维参数添加少量偏差, 使偏差在聚合算法的浮动范围内干扰全局模型收敛, 甚至可以在浮动范围内寻找合适的偏差实现后门攻击. Xie 等人[26]发现 Krum[14]和 Coordinate-wise Median[15]算法未考虑聚合梯度的方向, 因此巧妙构造恶意梯度, 使聚合梯度与正确梯度的内积为负数 (即方向相反), 破坏全局模型.

文献 [27–29] 则将模型投毒攻击转化为最优化问题进行求解. 其中 Bhagoji 等人[27]通过在前一轮的全局梯度上添加梯度 δ 来实施定向攻击, 其优化的目标函数是寻找使全局模型交叉熵损失最小的 δ. 另外, 考虑到聚合服务器可能使用验证数据集或者利用模型更新的统计学特征检测恶意梯度, 他们在目标函数中还引入了恶意方本地数据的训练损失, 以及 δ 和上一轮迭代中所有参与方模型更新平均值的 l_2 距离, 从而实现隐蔽的模型投毒攻击. Fang 等人[28]则是在前一轮全局聚合上添加干扰向量$-\lambda \boldsymbol{s}$, 其中 $\boldsymbol{s}$ 为本轮迭代中若无恶意方参与时全局模型参数的变化方向; λ 为比例系数, 用于放大恶意梯度的影响. 他们的目标是寻找最大的 λ, 使本轮聚合的全局模型尽量偏离前一轮聚合的变化方向, 实现非定向攻击. 考虑到 Krum[14]、Trimmed-mean[24]和 Median[15]安全聚合算法的存在, 他们将聚合算法转化为目标函数的约束条件进行求解. Shejwalkar 等人[29]则为模型投毒攻击设计了一个可适配不同安全聚合算法的通用框架. 他们也是通过添加干扰向量$-\lambda \boldsymbol{s}$ 实施非定向攻击, 但他们为 $\boldsymbol{s}$ 提供了 3 个选项: 正确梯度平均值的单位向量、正确梯度的标准差向量和正确梯度平均值的符号函数, 并通过实验证明不同的安全聚合算法适用不同的 $\boldsymbol{s}$. 他们根据聚合算法的工作原理, 将其转化为目标函数和约束条件后求解最大的 λ. 与文献 [28] 的工作相比, 他们还探讨了 AFA[38]和 Fang 等人[28]的其他安全聚合算法.

数据投毒和模型投毒都是通过上传恶意的模型更新破坏全局模型, 两者的区别在于数据投毒攻击不会干扰参与方的本地训练过程, 而模型投毒攻击可以跳过本地训练, 利用算法伪造任意模型更新. 因此, 模型投毒攻击不受

模型训练的限制, 威胁性更大, 但攻击难度也更高, 需要攻击者完全控制一个或多个参与方. 在当前联邦学习的应用中, 参与方的攻击成本较低, 导致模型投毒攻击大行其道, 加之其强大的破坏效果, 因此受到学术界的广泛关注.

2.2.3 后 门

后门 (backdoor) 攻击是在模型中埋藏一个后门, 攻击者可以通过预先设定的触发器 (trigger) 激活后门, 使模型对带有触发器的数据输出设定的标签, 同时不影响正常数据的推断. 例如, 后门攻击可以使自动驾驶模型正确识别普通的停车标志, 而将带有黄色方块的停车标志识别为限速标志, 此时黄色方块就是触发器[39]. 在集中学习中, 后门攻击通常是一种特殊的定向数据投毒攻击. 而在联邦学习中, 攻击者除了污染训练集外, 还可以上传恶意的模型更新植入后门, 因此联邦学习的后门攻击可以通过数据投毒或模型投毒实现, 三者的关系如图 4 所示.

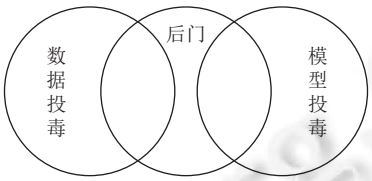

图 4 联邦学习中后门、数据投毒和模型投毒的关系

Nuding 等人[30]在交通标志分类的实验结果表明: 在联邦学习中可以通过数据投毒实现后门攻击, 而且增加恶意方比例和毒化训练数据量可以进一步提高后门攻击的成功率. Nguyen 等人[31]则证明了在基于联邦学习的物联网入侵检测系统中植入后门的可行性: 恶意方在网络中插入恶意流量, 并将恶意流量的比例控制在 10%–20% 的范围内, 可以在保证较高攻击成功率的同时又避免被检测到异常行为.

文献 [25,32–34] 都是通过模型投毒实现后门攻击. 在 Bagdasaryan 等人[32]的研究中, 恶意方先在本地训练带有后门的恶意模型, 然后上传恶意模型和前一轮全局模型的线性组合, 使平均聚合后的全局模型收敛为恶意模型, 但是他们并未验证攻击在 Krum[14]等安全聚合算法防护下的有效性. Sun 等人[33]与 Bagdasaryan 等人[32]的研究类似, 在其基础上考虑了正常参与方正确标记带有后门的数据样例的情况. 而 Baruch 等人[25]则借鉴 Bagdasaryan 等人[32]的设计, 通过在偏差浮动范围内寻找可植入后门的模型参数, 实现可攻破 Krum[14]等安全聚合算法的后门攻击. Xie 等人[34]则充分利用联邦学习的分布式计算特点, 设计分布式后门攻击 (distributed backdoor attack, DBA), 将触发器拆分成多个部分交由不同的恶意方植入, 有效地提升了触发器的攻击效果, 且可以绕过安全聚合算法的检测.

2.2.4 恶意服务器

聚合服务器负责全局模型的初始化、聚合和更新, 直接影响全局模型. 在目前的联邦学习架构中, 参与方在每轮迭代开始时都会使用聚合服务器下发的全局模型覆盖本地模型, 不会对全局模型的正确性进行检验, 因此恶意服务器可以跳过聚合过程直接下发恶意模型, 在参与方的本地模型植入后门, 给参与方的应用带来严重的威胁. 因为恶意服务器的攻击方法明显, 且服务器的安全防护措施较为完善、攻击成本高, 所以目前并没有相关的研究.

2.2.5 攻击方法总结

综上所述, 目前针对联邦学习的安全攻击方法及其威胁模型如表 1 所示. 另外, 表 1 还总结了每种攻击验证所使用的数据集, 包括图像领域的 CIFAR-10[40]、MNIST[41]、Fashion-MNIST[42]、AT&T[43]、FEMNIST[44]、Breast Cancer Wisconsin[45]、CH-MNIST[46]、European Traffic Signs[47]、EMNIST[48]和 Tiny-imagenet[49], 流量领域的 DIoT-Attack[50]、DIoT-Bengin[50]和 UNSW-Benign[51], 文本领域的 Reddit, 以及其他领域的 Adult Census[45]、Purchase[52]和 LOAN[53].

从表 1 可以推断目前联邦学习的安全攻击主要集中在图像领域, 对文本领域研究较少, 且没有音频方向的研究, 因此在文本和音频领域还有较大的研究空间. 另外, 目前研究的攻击类型主要是模型投毒攻击, 当联邦学习应用于不同组织间的协作时, 参与方攻击成本的增加将提高实施模型投毒的难度. 如何在安全聚合算法的保护下实施数据投毒攻击, 也是一个值得研究的问题.

表 1 联邦学习的安全攻击总结

文献	攻击类型	威胁模型			验证数据集
		攻击目标	攻击者知识	攻击者能力	
Tolpegin等人[21] (2020)	数据投毒	定向	—	控制参与方数据	CIFAR-10/Fashion-MNIST
Zhang等人[22] (2019)	数据投毒	定向	—	控制参与方: m=1	AT&T/MNIST
Zhang等人[23] (2021)	数据投毒	定向	—	控制参与方: m=1	CIFAR-10/Fashion-MNIST/MNIST
Mhamdi等人[24] (2018)	模型投毒	非定向	聚合算法/模型更新	控制参与方: $m\geqslant 1$, $m/n<0.5$	CIFAR-10/MNIST
Baruch等人[25] (2019)	模型投毒/后门	非定向/定向	数据分布	控制参与方: $m\geqslant 1$, $m/n<0.5$	CIFAR-10/MNIST
Xie等人[26] (2020)	模型投毒	非定向	聚合算法/模型更新	控制参与方: $m\geqslant 1$, $m/n<0.5$	CIFAR-10
Bhagoji等人[27] (2019)	模型投毒	定向	数据分布	控制参与方: $m\geqslant 1$, $m/n<0.5$	Adult Census/Fashion-MNIST
Fang等人[28] (2020)	模型投毒	非定向	—	控制参与方: $m\geqslant 1$, $m/n<0.5$	Breast Cancer Wisconsin/CH-MNIST/Fashion-MNIST/MNIST
Shejwalkar等人[29] (2021)	模型投毒	非定向	聚合算法/模型更新	控制参与方: $m\geqslant 1$, $m/n<0.5$	CIFAR-10/FEMNIST/MNIST/Purchase
Nuding等人[30] (2020)	后门	定向	—	控制参与方数据	European Traffic Signs
Nguyen等人[31] (2020)	后门	定向	—	控制参与方: m>1	DIoT-Attack/DIoT-Bengin/UNSW-Benign
Bagdasaryan等人[32] (2020)	后门	定向	—	控制参与方: m=1	CIFAR-10/Reddit
Sun等人[33] (2019)	后门	定向	—	控制参与方: $m\geqslant 1$	EMNIST
Xie等人[34] (2019)	后门	定向	—	控制参与方: m>1	CIFAR-10/LOAN/MNIST/Tiny-imagenet

注: (1) 攻击者知识中“模型更新”表示每轮迭代中所有参与方上传的模型更新, “数据分布”表示所有参与方本地训练集的数据分布; (2) 攻击者能力中m表示攻击者控制的参与方(即恶意方)的数量, n表示参与方总数

2.3 安全防护方法

针对联邦学习面临的安全威胁, 许多学者研究了各种防护方法以提高联邦学习的安全性. 根据防护方法采用的技术手段, 主要可分为以下 4 类: 设计安全聚合算法、结合区块链、利用安全硬件和加固模型, 此外还有个别文献提出一些特殊方法. 每类安全攻击可以采用不同的方法进行防御, 具体如表 2 所示. 注意, 因为后门可以通过数据投毒和模型投毒实现, 所以两者的防护方法也可用于防御相应的后门攻击, 因此表 2 不再单独列出后门攻击的防护方法.

表 2 联邦学习的安全攻击和防护方法

攻击手段	防护方法
数据投毒	设计安全聚合算法[14,15,21,24,28,29,54–72]、结合区块链[73–77]、加固模型[78,79]
模型投毒	设计安全聚合算法[14,15,24,28,29,54,55,57–72]、结合区块链[73–77]、利用安全硬件[80,81]
恶意服务器	结合区块链[73–76,82,83]、利用安全硬件[80]

2.3.1 设计安全聚合算法

通过分析联邦学习的攻击手段, 可以推断目前联邦学习的安全威胁主要来自恶意方, 但因参与方数量大、范

围广且难以控制, 所以现有安全防护方法的研究主要集中在服务器的聚合算法. 安全可靠的聚合算法可以保证即使联邦学习系统中存在恶意节点, 全局模型也能正确收敛. 目前安全聚合算法可分为以下两类: 基于模型更新特征差异的聚合算法和基于验证数据集的聚合算法. 在本节的后续表述中, 为方便说明, 设 n 和 m 分别表示参与方的总数和恶意方的数量.

(1) 基于模型更新特征差异的聚合算法

这类算法的主要思想是, 为了破坏全局模型, 攻击者上传的恶意模型更新不同于正常更新. 因此聚合服务器可以基于收集的所有模型更新从不同的特征分析更新间的差异, 区分恶意更新与正常更新, 或只选择恶意可能性较小的部分更新进行聚合.

文献 [14,15,24,54–58] 都是根据梯度的 l_p 距离来衡量两个梯度的差异, 都假定恶意梯度远离正常梯度的均值, 因此提出了一系列基于梯度平均值或中位数的聚合算法. 这些算法都是基于拜占庭将军问题的分析, 因此要求联邦学习网络中恶意方的数量不能超过正常参与方 (即 $n>2m$). 其中 Blanchard 等人[14]提出了 Krum 和 Multi-krum 算法: Krum 是从参与方上传的梯度中选择一个梯度作为全局模型的梯度, 要求该梯度在 l_2 范数空间与它的 $n–m–2$ 相邻梯度距离最近; Multi-krum 则是利用 Krum 算法依次选择 c 个参与方, 之后取它们的平均梯度作为聚合结果, c 的取值需要满足 $n–c>2m+2$. Chen 等人[54]则是将参与方平均分为 k 组, 计算每组的平均梯度后取这 k 个均值的几何中位数作为聚合结果, 其中 k 的取值与 m 相关. Mhamdi 等人[24]设计的聚合算法 Bulyan 与 Multi-krum 类似, 区别在于取的是 c 个参与方 ($c\leqslant n–2m$) 梯度的截尾平均数. Xie 等人[55]和 Yin 等人[15]都证明分别取梯度每一维的中位数作为全局模型对应维度的参数可以提高安全性. 另外, Yin 等人[15]还提出可以用截尾平均数 (截尾系数 α 需满足 $m/n\leqslant\alpha\leqslant 0.5$) 进行聚合. Cao 等人[56]则研究基于图的聚合算法: 首先将参与方作为图上的节点, 如果两个参与方梯度的 l_2 距离小于阈值则在这两点间添加一条链路; 之后服务器在生成的图上寻找最大团, 计算团上节点梯度的平均值作为聚合结果. Lu 等人[57]则计算参与方本地模型与上一轮全局模型的 l_2 距离, 并基于高斯分布的概率密度函数为参与方赋予不同的权重, l_2 距离越接近中心权重越高, 聚合结果为上传梯度的加权平均值. Wu 等人[58]则通过方差缩减技术 SAGA[84]减少随机梯度的噪声, 使恶意梯度更容易区分, 再结合中位数聚合算法, 进一步提高模型的安全性. 然而, Baruch 等人[25]证明了上述聚合算法的假定不一定成立, 且只适用于独立同分布 (independently identical distribution, IID) 的数据集, 无法应用于 Gboard 等联邦学习应用中, 因此上述聚合算法有较大的局限性.

文献 [59–61] 都是利用向量的余弦相似度 (cosine similarity) 描述恶意更新和正常更新的差异. Muñoz-González 等人[59]在每轮迭代中, 首先计算模型更新集合的加权平均值 (权重为参与方上传正常更新的概率); 之后计算每个更新与加权平均值的余弦相似度, 用于区分恶意方和正常参与方, 若余弦相似度超过一定范围则判断为恶意方, 将其移出本轮聚合; 最后取剩余参与方的加权平均值, 并基于参与方的判定结果利用贝叶斯模型更新参与方的权重. Khazbak 等人[60]则是根据参与方的模型更新与其他参与方的余弦相似度计算相似度分数, 分数越高说明和大部分参与方的模型更新越相近, 最后取分数最高的前 $n–m$ 个参与方进行聚合. Muñoz-González 等人[59]和 Khazbak 等人[60]的设计思路都是, 在正常参与方占主体 (即 $n>2m$) 的攻击场景中, 相比于恶意更新, 正常更新与其他更新较为相似. 而 Fung 等人[61]则放开对恶意方数量的限制, 只要求网络中至少有一个正常参与方, 即 $n–m\geqslant 1$. 他们利用在多个恶意方联合实施定向投毒的场景中, 恶意更新的多样性低于正常更新的特点, 提出防护算法 FoolsGold: 在每轮迭代中, 首先计算每个参与方的历史聚合更新, 之后根据其历史聚合更新与其他参与方的最大余弦相似度调整参与方的权重 (最大相似度越低, 恶意更新的可能性越高, 权重越小), 最后取加权平均值作为聚合结果.

此外, 还有部分学者提出了其他类型的区分特征, 如 Yang 等人[62]是根据正常梯度的 Lipschitz 特征非常接近真正梯度的特点, 设计基于中位数的聚合算法. 他们在每轮迭代中, 取 Lipschitz 特征在中位梯度一定范围内的梯度集合计算平均值, 但是该算法只适用于独立同分布的数据集. 而 Tolpegin 等人[21]是从高维的模型更新中提取特定数据标签的相关参数, 利用主成分分析 (principal component analysis) 对参数进行降维, 从而区分恶意更新和正常更新. Shejwalkar[29]则借鉴集中学习的数据投毒检测算法[85], 设计基于奇异值分解 (singular value decomposition) 的异常检测算法.

文献 [63,64] 则提出分组聚合的方法. Yu 等人[63]首先将所有模型更新用 K-means 算法[86]聚类分组, 随后在每个分组中使用其他安全聚合算法计算聚合梯度, 最后取每个分组聚合梯度的加权平均值 (权重为分组中参与方的数量). Singh 等人[64]则根据参与方的人口属性 (例如性别、年龄等) 进行分组. 在分组聚合方式中, 恶意方可能被分配到不同的组, 也可能集中在某个分组. 前者恶意方的影响会被分组内的安全聚合算法削弱, 而后者多个恶意方被压缩成一个恶意分组, 且分组权重也限制了其破坏性, 因此在一般应用场景下可以提高联邦学习的安全性. 但是, 攻击者可以反向利用分组逃避安全聚合算法的检查, 通过巧妙构造只有恶意梯度且梯度幅度较大的分组, 绕过分组内的安全聚合算法, 进而通过恶意分组威胁全局模型.

针对部分安全聚合算法只适用于独立同分布数据集的问题, He 等人[65]提出: 在服务器收集模型更新后, 利用无替换重采样技术平衡样本, 再与其他聚合算法相结合, 可以使这些聚合算法也适用于非独立同分布的场景.

(2) 基于验证数据集的聚合算法

基于验证数据集的聚合算法是利用验证数据集对参与方上传的模型更新进行验证, 根据参与方模型更新的准确性判断其是否为恶意更新. 其中文献 [28,66–70] 的研究都是由聚合服务器进行验证, 要求服务器拥有与参与方相似的数据样本.

Wang 等人[66]的研究是这类算法最基础的设计: 聚合服务器计算每个参与方上传梯度在验证数据集上的分类准确率, 如果准确率低于设定的阈值则归为恶意方并排除, 最终对筛选后的参与方求平均值. Tan 等人[67]沿用类似的思路, 并在服务器端增加一个深度强化学习模型, 可以基于参与方的历史行为和本轮验证结果, 指导服务器在下轮迭代中选择合适的参与方进行聚合, 降低训练开销. Chen 等人[68]则采用分组验证的思路: 在每轮迭代中, 他们首先依据模型更新的 l_2 距离, 采用 K-means 算法[86]对参与方的模型更新进行分组; 之后用测试数据集对每个聚类进行验证, 如果准确率低于阈值则排除该聚类下的所有模型更新.

文献 [28,69,70] 的研究不再单纯依赖验证准确率来评估模型更新的可信度, 而是设计了不同的评价指标. 其中 Xie 等人[69]提出的 Zeno 聚合算法的评价指标是随机下降分数 (stochastic descendant score), 该分数包含两部分: 模型更新导致的损失函数下降和模型更新的幅度. 损失函数下降越多 (即全局模型可以更快收敛), 幅度越小 (即对全局模型的影响较小), 分数越高. 在每轮迭代中, Zeno 利用验证数据集计算每个参与方的随机下降分数, 之后取分数最高的前 $n-m$ 个参与方的模型更新的平均值作为聚合结果.

Cao 等人[70]则提出以聚合服务器自身训练结果为信任根, 将服务器和参与方模型更新的相似度作为评价指标. 在每轮迭代中, 服务器使用验证数据集训练全局模型. 在聚合时, 服务器首先计算自己和参与方模型更新的余弦相似度, 并使用 ReLU 函数整流后作为参与方的信任分; 之后标准化参与方模型更新的幅度; 最终以信任分为权重, 计算标准化后的模型更新的加权平均值. 信任分越高表示参与方的模型更新与服务器的训练结果越接近, 即可信度越高, 权重越大.

Fang 等人[28]则借鉴集中学习数据投毒的防护方法 RONI[87]和 TRIM[88], 提出一种可以与其他聚合算法相结合的辅助措施. 针对每个参与方模型更新 w, 他们利用验证数据集分别计算全局模型聚合 w 和不聚合 w 的验证错误率, 将两者的差值作为参与方的得分, 得分越高说明 w 给全局模型造成的负面影响越大. 他们移除分数最高的前 m 个参与方后, 可结合其他聚合算法对剩余的参与方进行聚合. 另外, 他们也提出了其他评分指标, 包括模型的损失函数, 以及错误率和损失函数的联合值, 并通过实验证明采用损失函数或联合值作为分数指标可以增强防护效果.

上述聚合算法都是假定服务器拥有验证数据集, 而针对服务器无法事先收集数据样本的问题, Zhao 等人[71]通过在服务器部署 GAN, 将每轮全局模型作为 GAN 的判别网络 D, 利用 GAN 的生成网络 G 生成和参与方相似的数据样例作为审计数据集. 生成的审计数据集即可用于检查参与方模型的准确性.

不同于服务器验证的思路, Zhao 等人[72]实现参与方的交叉验证. 在每轮迭代中, 聚合服务器为每个模型更新添加高斯噪声, 并委派多个参与方利用他们的本地数据对更新进行评估, 之后综合参与方的评估结果调整更新的权重, 最终计算加权平均值.

2.3.2 结合区块链

联邦学习作为一种分布式计算框架, 与分布式账本-区块链 (blockchain) 技术高度契合, 因此部分学者提出了

将联邦学习与区块链相结合的防护方案, 利用区块链的安全特性提高联邦学习系统的安全性.

在 Bao 等人[73]设计的区块链 FLChain 中, 区块保存参与方的个人信息、数据资源描述、历史联邦训练活动和可信度, 其中历史联邦训练活动指的是参与方在每轮迭代中上传的本地梯度和聚合的全局模型. 在 FLChain 中, 参与方通过注册加入区块链, 在训练中将本地梯度上链, 并下载其他参与方的梯度进行聚合. FLChain 会选择可信度最高的参与方作为领导, 将其聚合的模型作为新的全局模型以达成共识. 参与方可以依据全局模型对其他参与方进行评估, 更新其可信度. 参与方的相互审计可以及时发现恶意方的存在, 缓解投毒攻击的影响.

Li 等人[74]提出的 BFLC 设计了两种区块, 分别保存全局模型和本地模型更新. BFLC 通过部分参与方组成的委员会达成共识. 在每轮迭代中, 除委员会外的其他参与方将本地模型更新发送给委员会, 委员会所有成员将本地数据作为验证数据集, 用验证准确率的中位数作为参与方的分数, 分数高的模型更新才能上链. 当链上有效的模型更新区块数量达到阈值后, 委员会执行聚合过程, 将新的全局模型上链, 并根据其他参与方在上一轮训练的分数重新选举委员会成员. 委员会交叉验证的方式可以有效防护恶意方的投毒攻击. Peng 等人[82]也是通过委员会检验和聚合参与方的模型更新, 不同的是他们设计的区块中只保存全局模型的可验证证明 (verifiable proofs), 支持参与方对全局模型进行审计. 他们还提出一种新的区块链认证数据结构, 用于提高可验证证明的搜索效率, 以及支持委员会的安全轮换. 而 Shayan 等人[75]提出的 Biscotti 则包含两种委员会: 验证委员会和聚合委员会, 分别负责参与方模型更新的检测和聚合. Biscotti 的区块中保存了全局模型 w_t、模型更新的聚合值Δw 和每个更新的多项式承诺 (polynomial commitments)[89], 其中多项式承诺可以提供参与方隐私保护和聚合结果的可验证性. Biscotti 采用 PoF (proof of federation) 共识算法.

Qu 等人[83]为工业 4.0 网络中的认知计算提出基于区块链的联邦学习框架. 他们将中心服务器替换为由多个矿工组成的区块链, 矿工负责收集工业 4.0 设备上传的模型更新并生成区块, 通过 PoW (proof of work)[90]共识算法选出获胜者, 由获胜者聚合模型后上链. Zhao 等人[76]也为物联网智能家居场景设计了类似的框架, 但他们采用的是 Algorand[91]共识算法, 从矿工中选出一个领导负责聚合并更新全局模型.

Liu 等人[77]为 5G 网络设计的安全联邦学习框架则保留了聚合服务器, 他们将联邦学习与以太坊相结合, 使用以太坊的 PoS (proof of stake)[92]共识算法和智能合约. 聚合服务器在下发联邦学习任务时会发布一个智能合约, 该合约的内容是利用测试数据对模型更新进行验证. 以太坊的矿工会利用智能合约验证参与方上传的模型更新, 只有验证合格的更新才会发给服务器进行聚合.

表 3 从区块链的组成节点、采用的共识算法、区块保存的数据内容和负责模型聚合的节点这 4 个维度对基于区块链的联邦学习框架进行总结.

表 3　基于区块链的联邦学习框架总结

文献	区块链节点	共识算法	区块保存的数据	聚合节点
Bao等人[73] (2019)	参与方	基于可信度的领导选举	参与方的个人信息、数据资源描述、历史联邦训练活动、可信度	可信度最高的节点
Li等人[74] (2021)	参与方	委员会共识	本地模型更新或全局模型	委员会
Peng等人[82] (2021)	参与方	多数表决	全局模型的可验证证明	委员会
Shayan等人[75] (2021)	参与方	PoF	全局模型w_t、模型更新的聚合值Δw、每个更新的多项式承诺	聚合委员会
Qu等人[83] (2021)	矿工	PoW	本地模型更新、全局模型	PoW获胜的矿工
Zhao等人[76] (2021)	矿工	Algorand	本地模型更新、全局模型	矿工领导
Liu等人[77] (2020)	矿工	PoS	本地模型更新、全局模型	聚合服务器

与其他安全防护方法相比, 将联邦学习与区块链相结合具有以下优势.

(1) 利用区块链去中心化的特点, 即不依赖中心管理节点实现数据的分布式记录、存储和更新, 可以移除联邦学习系统中的聚合服务器, 转而从所有参与方节点中选择单个或多个节点执行聚合操作, 并在所有节点间取得共识, 从而防御恶意服务器的攻击[73–76,82,83].

(2) 通过将模型更新和全局模型上链, 使所有参与方都可以对其进行检验, 及时发现恶意行为. 现有基于验证数据集的安全聚合算法主要是由聚合服务器执行, 要求服务器事先准备验证数据集, 但这个条件在实际应用中存在一定的局限性: 一方面服务器无法接触参与方的数据, 其拥有的验证集可能与实际的数据分布有明显差异; 另一方面当联邦学习应用于数据非独立同分布的场景时[3], 服务器的数据无法涵盖所有参与方的数据分布. 上述两个因素导致服务器主导的检验容易出现误判, 且可能产生不公平现象, 即少数参与方经常被忽略. 通过结合区块链, 联邦学习的每个参与方都可以直接或间接地参与到模型检测和聚合的过程中, 从而及时发现系统中的恶意节点, 提高异常检测的准确率.

2.3.3 利用安全硬件

针对模型投毒跳过本地训练直接上传恶意更新的攻击行为, 部分学者提出可利用安全硬件保证联邦学习本地训练的完整性, 防止训练过程受攻击者干扰. 他们主要是利用可信执行环境 (trusted execution environment, TEE) 进行防护, 如 Intel 的 SGX[93]等. SGX 是 2013 年 Intel 推出的一套指令集扩展, 它以硬件安全为强制性保障, 不依赖于固件和软件的安全状态, 提供用户空间的可信执行环境, 通过一组新的指令集扩展与访问控制机制, 实现不同程序间的隔离运行, 保障用户关键代码和数据的机密性与完整性不受恶意软件的破坏[94]. SGX 提出一种安全容器 enclave[95], 用于存放应用程序的敏感数据和代码. SGX 允许应用程序指定需要保护的敏感数据和代码, 并将其加载到 enclave 中, 之后 SGX 会保护它们不被外部软件所访问[94].

Chen 等人[80]将参与方的本地模型训练和服务器的聚合过程都加载到 TEE 的 enclave 中执行, 且参与方和聚合服务器间的模型交互也是经由 enclave 间的安全通道完成. 一方面保证本地训练过程的完整性, 避免攻击者跳过或干扰本地训练, 上传伪造的模型更新; 另一方面防止恶意服务器无视参与方上传的更新, 直接下发恶意模型. Zhang 等人[81]也提出利用 TEE 保护联邦学习的本地训练, 同时他们还考虑到 TEE 训练模型导致的性能下降问题. 目前 SGX 只支持 CPU, 无法应用于 GPU 加速训练的深度学习模型, 而且有限的 PRM (processor reserved memory) 会导致频繁的页切换, 这些因素都会明显降低本地训练的效率. 于是, 他们提出在 GPU 训练本地模型的过程中随机选择几轮训练在 TEE 中执行, 利用 TEE 对训练的完整性和正确性进行验证, 从而削弱 TEE 对本地训练效率的影响.

虽然利用安全硬件可以有效防护模型投毒攻击, 但是无法保证本地训练集的可靠性, 因此不适用于防护数据投毒攻击.

2.3.4 加固模型

针对数据投毒攻击, 部分学者借鉴集中学习的防御思路, 提出通过修改联邦学习模型的结构来提高模型的健壮性, 降低污染数据的危害程度.

Zhao 等人[78]在模型训练过程中引入增强模型稳定性的操作 (如 dropout、权重衰减和梯度截断等), 提高模型的泛化能力, 并通过实验证明增加 dropout 操作可以有效降低后门攻击的成功率. Zhang 等人[79]则发现恶意梯度的多样性与正常梯度有明显差异, 以此提出基于关键学习 (pivotal learning)[96]的防护方法: 联邦对抗训练 (federated adversarial training, FAT). FAT 在服务器端增加一个分类模型 f_2, 负责推断上传梯度属于哪个参与方. 为提高模型的健壮性, 需要尽可能模糊恶意梯度和正常梯度的分类边界, 即尽量降低 f_2 的准确率. 为此, FAT 修改模型的最优化公式和损失函数, 在每轮迭代中将 f_2 的损失添加到模型的目标函数中下发给参与方进行训练, 使收敛后的全局模型可以抵御污染的数据样本.

Ibitoye 等人[97]则提出针对联邦学习对抗样本 (adversarial samples)[98]攻击的防护方法 DiPSeN. 对抗样本攻击是指攻击者在模型的输入样本中故意添加细微干扰, 使模型输出错误的推理结果. DiPSeN 通过在神经网络模型的结构中添加一层满足差分隐私的噪声和一层 SELU 激活函数, 提高模型的鲁棒性.

2.3.5 其 他

除了上述 4 种主要的安全防护技术手段外, 部分学者也提出一些其他方法防御安全攻击. 其中包括如下几种.

Chang 等人[99]认为参与方和聚合服务器分享模型参数是一种错误的设计, 因此提出一种新的联邦学习框架 Cronus, 利用数据标签替代模型参数, 实现参与方模型的知识转移. 他们假定存在一个可以被所有参与方访问的公共数据集. 在每轮迭代中, 参与方首先用本地数据训练模型, 之后用本地模型对公共数据集进行预测, 将预测的数据标签上传给聚合服务器; 随后服务器聚合参与方的预测标签, 并下发聚合结果; 最终参与方结合本地数据和带有聚合标签的公共数据集重新训练, 更新本地模型. 在 Cronus 框架中, 参与方的本地模型都是通过本地训练得到, 可以有效削弱恶意服务器和恶意方对正常参与方本地模型的影响.

Kang 等人[100]针对联邦学习系统中存在恶意方的安全问题, 为联邦学习引入信誉机制, 即根据参与方在联邦学习任务的表现为其设定一个信誉值, 服务器优先选择信誉好的参与方进行聚合. 在联邦学习的每轮迭代中, 服务器会利用 FoolsGold[61]等方法对参与方上传的模型更新进行检测, 根据参与方的行为更新其信誉. 信誉值可以供后续其他任务的服务器参考. 为避免参与方的信誉被攻击者篡改, 他们采用区块链保存参与方的信誉值, 保证数据安全.

Guo 等人[101]为了避免参与方盲目采用恶意服务器下发的全局模型, 提出基于同态哈希 (homomorphic hash) 函数的可验证的聚合协议 VERIFL, 使参与方在每轮迭代结束后可以对全局模型的准确性进行检查. 参与方在上传模型更新的同时计算更新的哈希值. 等服务器下发聚合模型后, 参与方向其他参与方请求哈希值, 利用同态哈希验证聚合结果是否正确. 然而, 因同态哈希函数的限制, VERIFL 只能采用线性聚合算法, 无法防御投毒攻击.

2.4 与集中学习的共性和差异

本节从威胁模型、安全攻击手段和安全防护方法这 3 个方面对集中学习和联邦学习在安全问题上的共性和差异进行分析.

2.4.1 威胁模型的异同

集中学习和联邦学习在威胁模型上存在以下共同点: 首先, 集中学习和联邦学习的攻击目标是一致的, 都是尽可能破坏模型的可用性, 使模型在推理阶段无法正常工作, 具体的攻击目标都包括非定向攻击和定向攻击. 其次, 在攻击者能力方面, 研究两者的安全攻击时都会考虑攻击者是否可以控制训练集.

对于威胁模型的其他方面, 集中学习与联邦学习有如下差异.

(1) 攻击者知识

因为集中学习的模型通常是由服务提供商在本地训练后开发给用户使用, 攻击者无法轻易接触目标模型, 所以在集中学习中攻击者知识指的是攻击者对于目标模型的了解程度, 可分为黑盒攻击和白盒攻击[102]. 其中白盒攻击是指攻击者可以获取目标模型的所有信息, 包括模型结构和训练参数等. 而在黑盒攻击中, 攻击者只能利用服务提供商开放的接口查询目标模型对特定输入的输出结果, 无法获得目标模型的相关信息, 在实际应用中更为常见. 而联邦学习现有的安全研究主要探讨来自系统内部的威胁, 即聚合服务器或参与方, 因两者都会对目标模型进行计算, 所以联邦学习面临的安全攻击默认都是白盒攻击. 在联邦学习中, 攻击者知识主要是指攻击者对目标联邦学习系统的背景知识, 如聚合算法、其他参与方的模型更新等.

(2) 攻击者能力

除了控制训练数据集外, 部分集中学习的安全研究会假定攻击者可以修改训练好的目标模型, 攻击者通过修改模型的权重[103]或者在现有模型中插入新的模块[104,105]发动攻击. 因为集中学习的模型通常是在服务器上训练, 所以攻击能力不包括参与模型训练. 而在联邦学习中, 因为参与方的攻击成本较低, 在实际应用中攻击者可以控制参与方的训练, 所以相关的研究会将其纳入考虑范围. 另外, 因为联邦学习是由多个参与方协作完成, 恶意方的数量与攻击效果密切相关, 所以在联邦学习相关的安全研究中需要明确说明恶意方的数量要求.

2.4.2 安全攻击手段的异同

集中学习和联邦学习都包括训练阶段和推理阶段, 两者的区别只在于训练阶段. 表 4 对集中学习和联邦学习

的安全攻击手段的共性与差异进行总结. 从表 4 可以看出, 两者在推理阶段面临着相同的安全攻击——对抗样本攻击. 攻击者通过特意构造模型的输入样本, 使模型做出错误的推理, 一般是通过在模型输入上添加精心设计的扰动实现. 因为对抗样本攻击只修改测试数据, 所以适用于集中学习和联邦学习. 而在训练阶段, 集中学习和联邦学习都可能遭受数据投毒攻击和后门攻击. 因为集中学习的安全攻击都是针对单个目标模型, 所以可以将其扩展用于联邦学习的全局模型.

然而, 联邦学习中参与方和服务器协作的机制可以削弱集中学习的攻击效果, 但也引入了新的安全问题与挑战. 一方面, 聚合操作会降低单一或少量攻击者对全局模型的影响, 集中学习的数据投毒对全局模型的危害有限, 因此在研究联邦学习的安全攻击时, 需要解决聚合后攻击的有效性问题. 另一方面, 引入脆弱的参与方为攻击者干预模型训练创造了条件, 攻击者可以通过模型投毒破坏全局模型. 除了恶意方, 联邦学习的安全研究还需要考虑恶意服务器的存在. 此外, 集中学习与联邦学习在后门攻击的实施策略也存在差异. 在联邦学习中, 攻击者是通过数据投毒或模型投毒植入后门, 而在集中学习的后门攻击研究中, 虽然绝大部分都是基于数据投毒实现, 但也有部分学者研究其他植入后门的策略[106]. 这类攻击通常发生在模型的部署阶段, 攻击者直接修改训练后的目标模型, 通过修改模型权重[103]或在模型中插入隐蔽的模块[104,105]实施后门攻击.

表 4 集中学习与联邦学习的安全攻击手段的对比

类型	训练阶段					恶意服务器	推理阶段
	数据投毒	模型投毒	后门				对抗样本攻击
			数据投毒	模型投毒	修改模型		
集中学习	√		√		√		√
联邦学习	√	√	√	√		√	√

2.4.3 安全防护方法的异同

在安全防护方法上, 集中学习和联邦学习存在以下共同点: 对抗样本攻击的防护方法都适用于集中学习和联邦学习; 目前, 集中学习针对数据投毒的防御主要是使用鲁棒学习和数据清理来净化训练样本[107], 因此可以直接应用于联邦学习的参与方, 从而对模型进行加固.

而两者在安全防护方法上的差异具体如下.

(1) 在参与方为资源受限设备的联邦学习应用中 (如物联网等), 直接采用集中学习的安全防护方法可能会给设备带来一定的计算压力, 因此需要调整现有的防护方法, 在资源开销和安全性之间进行平衡, 以适用于联邦学习的场景.

(2) 与集中学习相比, 联邦学习面临着恶意参与方的威胁, 可以通过研究安全可靠的聚合算法进行防御. 此外, 也可以利用安全硬件避免训练过程受攻击者干扰, 提高联邦学习的安全性.

(3) 虽然集中学习和联邦学习都可以通过验证数据集检测模型的异常行为, 但是与集中学习完全掌握训练数据信息不同, 联邦学习的聚合服务器拥有的辅助数据可能和参与方的训练数据有明显差异, 导致这一防护方法在联邦学习中具有一定的局限性, 因此部分学者提出将联邦学习与区块链相结合, 让每个参与方都可以对全局模型进行检测.

3 隐私风险与保护

根据机器学习隐私保护的内容, 可将机器学习隐私分为训练数据隐私、模型隐私与预测结果隐私[108]. 对于模型隐私, 因为联邦学习需要参与方在本地训练模型, 模型算法、神经网络结构和参数等模型信息对参与方都是可见的, 所以联邦学习通常不考虑模型隐私泄露的风险. 而对于预测结果隐私, 集中学习和联邦学习面临的攻击手段和防护方法是一致的, 不在本文的讨论范围内. 因此本文对于隐私风险的总结和分析主要是针对训练数据隐私, 下文如无特殊说明, 隐私均指代训练数据隐私.

虽然联邦学习通过参与方和服务器交换模型参数的方式保护了参与方的本地数据, 但是学者研究发现交换的

模型梯度也可能泄露训练数据的隐私信息[109,110]. 对于集中学习, 模型倒推 (model inversion) 攻击可以从模型中反推训练数据的属性值[111], 这同样也适用于联邦学习的全局模型. 而联邦学习的训练机制也为隐私引入了新的风险.

(1) 联邦学习的模型信息对攻击者是可见的, 攻击者可以实施白盒隐私攻击.

(2) 联邦学习的训练包含多轮迭代, 攻击者可以利用模型在迭代过程的变化挖掘更多的数据信息.

(3) 攻击者可以通过参与方或服务器干扰模型训练过程, 修改模型参数, 使正常参与方在后续迭代中暴露更多本地数据信息.

因此许多学者专门针对联邦学习存在的隐私风险与保护方法进行研究. 本文以联邦学习面临的隐私攻击的发生逻辑和顺序对目前主要研究的攻击手段进行分类 (如图 5 所示), 具体分为成员推断攻击[112−115]、属性推断攻击[16,112,116−122]和窃听. 根据攻击者角色的不同, 隐私攻击发生在联邦学习的不同阶段, 如服务器是在模型聚合阶段发动隐私攻击. 注意, 在图 5 的推理阶段可以实施集中学习的隐私攻击手段, 这部分不在本文讨论范围内.

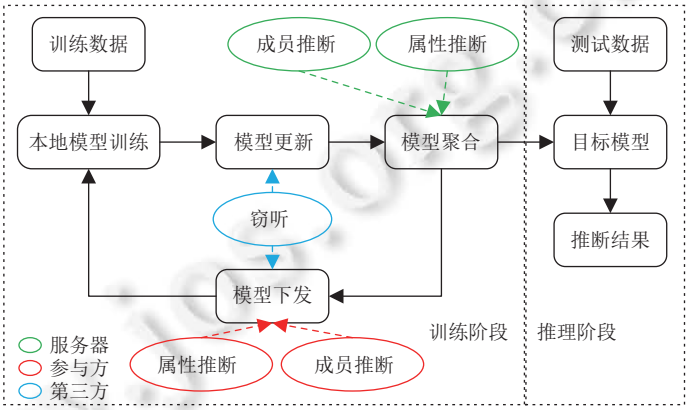

图 5 联邦学习面临的隐私攻击

3.1 威胁模型分析

本文从攻击者角色、攻击者目标、攻击者知识和攻击模式这 4 个维度对隐私攻击的威胁模型进行分析.

(1) 攻击者角色

攻击者角色是指攻击者在联邦学习系统中扮演的角色, 具体包括: 服务器、参与方和第三方. 其中服务器的攻击目的是提取与参与方训练数据相关的信息, 可以对单个参与方实施攻击. 而参与方是为了窃取其他参与方的训练数据隐私, 但因为参与方只能接触全局模型, 所以无法攻击特定的参与方. 第三方则是指没有参与到联邦学习训练过程的个人或组织, 他们只能通过窃听服务器和参与方的通信, 或者使用训练好的全局模型等方法推断联邦学习的模型信息或参与方的数据信息.

(2) 攻击者目标

攻击者的目标是从联邦学习的训练过程中提取参与方本地数据的隐私信息, 根据其具体目标可分为两类: 成员推断 (membership inference) 和属性推断 (property inference). 其中成员推断是推断某个数据样本是否在参与方的训练数据集中. 作为一个决策问题, 成员推断攻击的结果是输出某个数据样本属于参与方训练集的概率[109]. 而属性推断攻击的目的是推断训练数据的属性, 属性的概念比较广泛, 既可以是与模型主任务相关的属性 (如人种识别模型的训练照片里人物的肤色), 也可以是无关的属性 (如人种识别模型的训练照片里人是否配戴眼镜). 根据推断的目标属性, 攻击结果会呈现不同的形式, 可以输出训练数据拥有目标属性的比例, 甚至还可以利用推断的属性重构与训练数据相似的数据样本.

(3) 攻击者知识

攻击者知识是指攻击者对目标联邦学习系统所了解的背景知识, 在隐私攻击中要求的知识只有辅助数据集. 辅助数据集需要和参与方的本地数据相似, 且带有正确的主任务标签或属性标签.

(4) 攻击模式

攻击模式分为主动攻击和被动攻击. 其中主动攻击是指攻击者干扰联邦学习的正常流程, 如控制服务器跳过聚合过程下发恶意模型等, 而被动攻击是指攻击者不干预联邦学习, 只在服务器或参与方部署额外程序, 基于现有的数据和模型进行攻击.

3.2 隐私攻击手段

3.2.1 成员推断

在联邦学习中, 成员推断攻击是指攻击者利用参与方的模型更新或全局模型推断参与方的训练数据集中是否包含某个数据样本的攻击方法.

Melis 等人[112]利用模型嵌入层 (embedding layer) 的非零梯度可以反映什么单词出现在训练集的特点进行成员推断. 恶意方首先通过全局模型嵌入层的梯度变化收集词汇表序列 $[V_1, \ldots, V_T]$, 之后测试目标文本的单词集合 V_r 是否出现在序列中 ($V_r \subseteq V_t$), 从而判断目标文本是否为训练数据成员. 他们的攻击方法有很大的局限性, 只适用于带有嵌入层的模型 (如自然语言处理), 且推断过程中未考虑目标文本单词的排列顺序, 误报率较高.

Nasr 等人[113]发现训练数据会在模型损失函数的梯度留下可区分的足迹, 以此设计一个成员推断的深度学习模型. 该攻击模型以目标数据的标签、损失函数, 以及数据在目标模型每一层隐藏层 (hidden layer) 的梯度和输出为输入, 并用卷积神经网络 (convolutional neural network, CNN) 和全连接网络 (fully connected network, FCN) 分别提取隐藏层梯度和输出中与目标数据相关的特征, 组合进入一个全连接编码器 (fully connected encoder), 最终输出反映目标数据的成员信息嵌入的单一值, 该值可以表示目标数据为训练集成员的概率. 针对联邦学习经过迭代产生多个时间版本模型的情况, 他们将 CNN 和 FCN 组件的输入修改为目标数据在不同版本模型的隐藏层梯度和输出的叠加向量, 通过模型版本的变化学习更多目标数据的特征. 另外, 他们还提出一种主动攻击方法: 服务器或恶意方主动增加模型在目标数据的梯度. 如果目标数据为训练集成员, 正常参与方会在后续迭代中明显下降模型损失函数在目标数据的梯度, 成员推断模型可以检测到这种变化, 从而提高推断攻击的成功率.

文献 [114,115] 都是利用 GAN 和分类模型实现成员推断攻击. 其中 Chen 等人[114]提出的攻击只适用于参与方的数据标签不重叠的联邦学习场景中, 且开始训练前要求每个参与方声明其拥有的数据标签. 他们首先利用 GAN 生成不同标签的数据样例, 用数据样例和对应的标签训练一个分类模型. 该分类模型以目标数据为输入, 如果输出的标签与某个参与方声明的数据标签一致, 则判断目标数据为该参与方训练集的成员, 从而实现针对特定参与方的成员推断攻击. 而 Zhang 等人[115]设计的分类模型是通过学习真实标签周围的预测值分布, 区分目标模型的成员数据和非成员数据. 他们的攻击模型是以目标数据的预测值和标签为输入. 考虑到恶意方的训练数据有限, 他们利用 GAN 生成分类模型的训练数据样例, 提升攻击效果.

3.2.2 属性推断

属性推断攻击是推断参与方训练数据的敏感隐私属性, 包括模型任务相关属性和无关属性.

(1) 相关属性推断

模型任务相关属性是描述训练数据中每类数据的关键特征, 通过推断相关属性可以重构每类标签的训练数据, 因此这种攻击也可称为数据重构 (data reconstruction) 攻击. 重构的数据并不是真正的训练数据, 只是与训练数据相似的数据样本[112]. 目前实现数据重构攻击的技术思路主要包括两种: 利用 GAN 重构数据和将攻击转化为最优化问题求解.

文献 [116–118] 都是利用 GAN 实施数据重构攻击. 其中 Hitaj 等人[116]通过在恶意方部署 GAN 重构其他参与方特定标签的代表数据. 恶意方在每轮迭代中, 以全局模型作为 GAN 的判别网络 D, 利用生成网络 G 生成目标标签的代表数据. 为了提高攻击效率和增强攻击效果, 在前几轮迭代中, 恶意方为 GAN 生成的数据标记错误的标签, 训练后毒化全局模型, 诱导其他参与方在后续迭代中暴露更多目标标签数据的信息. 然而, 这种主动攻击会降低全局模型的准确性, 可能被检测到异常行为并排除. Wang 等人[117]则提出服务器可以利用 GAN 重构特定参与方的训练数据. 他们通过在服务器侧部署多任务 GAN (multi-task GAN) 学习目标参与方的数据分布, GAN 的判别网

络 D 需要同时完成 3 个任务: 区分输入数据的标签、真实性和对应参与方. 在 D 中引入数据参与方的识别可以使 G 生成针对特定参与方的数据样例, 但因为服务器无法接触参与方的本地数据, D 的数据参与方识别任务缺乏训练样本, 所以他们利用模型更新生成粗略的参与方代表数据对 D 进行训练. 而 Song 等人[118]在 Wang 等人[117]工作的基础上进一步扩展, 研究在参与方匿名的联邦学习场景中如何将模型更新与参与方绑定, 以便实施后续的数据重构攻击. 他们攻击的主要思路是, 模型更新可以反映参与方的数据分布, 该分布在迭代过程中是相对固定的, 且不同于其他参与方. 他们利用卷积孪生网络 (convolutional Siamese network) 计算不同迭代中两个模型更新生成的代表数据的相似程度, 判断两个模型更新是否属于同一参与方. 通过反复比对更新的相似程度最终将所有迭代中的模型更新按参与方进行分组. 虽然上述基于 GAN 的数据重构攻击在 MNIST[41]等数据集上效果显著, 重构的数据与真实的训练数据非常相似, 但是只适用于目标标签的数据成员相似的场景 (如 MNIST 数据集中数字 9 的手写图片在视觉上相似), 在其他应用场景中危害有限, 例如在性别分类模型中 GAN 为“女性”标签重构的数据只是一般的女性脸部图像, 无法在训练集中找到对应的图片[112].

Zhu 等人[16]则是将数据重构攻击转化成最优化问题进行求解. 他们利用模型梯度泄露训练数据信息的原理[112], 推论出如果重构数据可以使全局模型产生和参与方梯度相近的梯度信息, 则重构数据也和参与方的训练数据相似. 因此他们只需要寻找重构数据 (x, y), 最小化全局模型在 (x, y) 的梯度与目标梯度的 l_2 距离, 其中 x 为输入数据, y 为数据标签. 他们通过标准正态分布初始化 x 和 y, 并使用标准梯度下降法进行求解, 最终通过实验证明重构图片与训练图片在视觉上非常相似. 然而, 他们提出的 DLG 攻击效率较低, 在部分场景下重构的数据质量很差, 且只可用于重构小尺寸的图像, 因此文献 [119–121] 对其进行改进. 其中 Geiping 等人[119]采用梯度的角距离衡量梯度的相似程度, 将目标函数改为最小化全局梯度和参与方梯度的余弦相似度, 并基于梯度的符号函数采用 Adam 算法[123]求解最优化问题, 从而在多图像重构任务和大尺寸图像数据集上获得很好的攻击效果. Zhao 等人[120]则是对 DLG 中数据标签 y 的初始化方法进行改进. 他们从参与方梯度中提取样本的真实标签 y, 将最优化问题简化成只求解 x, 从而提高攻击效率和增强攻击效果. Wei 等人[121]则在目标函数中增加基于标签的 l_2 正则化项以增加攻击的稳定性, 同时将 (x, y) 的初始化方法、攻击终止条件和最优化方法作为配置项, 通过实验评估不同配置对攻击效果的影响.

(2) 无关属性推断

任务无关属性是指训练数据中对模型任务不起作用的特征信息, 理论上模型不应该泄露这类隐私, 这纯粹是模型训练过程的产物[112], 因此无关属性推断也称为无意识的特征泄露 (unintended feature leakage). 无意识的特征泄露不易察觉且难以检测, 且可能带来严重的隐私风险, 因此引起了部分学者的重视. 这类攻击没有明确的指向性, 具体的攻击目标因人而异, 例如 Melis 等人[112]的攻击目标是推断其他参与方的训练数据中是否拥有攻击者关心的属性, Shen 等人[122]则希望在保证主任务性能的前提下, 推断训练数据具有目标属性的参与方集合. Melis 等人[112]为参与方实施属性推断攻击提出被动和主动两种模式: 在被动攻击中, 他们首先计算全局模型在辅助数据集上的梯度, 并根据辅助数据是否具有目标属性贴上相应的标签, 随后用梯度和标签训练一个二分类器, 最终以参与方的模型更新为输入进行分类, 推断参与方的训练数据是否具有目标属性. 而在主动攻击中, 他们利用多任务学习 (multi-task learning) 提升攻击效果. 多任务学习是同时考虑多个相关任务的学习过程, 目的是利用任务间的内在关系提高单个任务学习的泛化性能[124]. 在本地训练时, 他们将模型损失函数修改为主任务和属性识别任务的联合损失, 促使全局模型学习目标属性的数据表现, 从而提高攻击的成功率. Shen 等人[122]则在服务器侧部署分类器, 在每轮迭代中首先使用辅助数据训练分类器, 该分类器以聚合后的模型更新为输入, 输出本轮聚合的参与方集合具有目标属性的可能性, 之后选择其他参与方进行下一轮迭代. 在遍历所有参与方后, 取可能性最高的参与方集合作为最终的推断结果. 此外, 他们还设计动态的参与方选择算法来提高攻击效率.

3.2.3 窃　听

窃听发生在参与方和服务器交互的过程中, 如果参与方和服务器之间是明文通信, 或者采用脆弱的加密通信方法, 攻击者就可以通过窃听获取参与方上传的模型更新以及服务器下发的全局模型, 进而实施隐私攻击. 窃听为联邦学习的第三方提供了窃取隐私的渠道.

3.2.4 攻击方法总结

综上所述, 目前针对联邦学习的隐私攻击方法及其威胁模型如表 5 所示. 另外, 表 5 还总结每种攻击验证时使用的数据集, 包括图像领域的 CIFAR-100[40]、CIFAR-10[40]、MNIST[41]、AT&T[43]、LFW[125]、FaceScrub[126]、PIPA[127]、BERT[128]、SVHN[129]、ImageNet[130]、CASIS-WebFace[131]和 CelebA[132], 文本领域的 Yelp-health[133]和 Yelp-author[133], 以及其他领域的 Purchase[52]、FourSquare[134]、Texas100[135]和 CSI[136].

表 5 联邦学习的隐私攻击方法总结

文献	威胁模型				验证数据集
	攻击者角色	攻击目标	攻击者知识	攻击模式	
Melis等人[112] (2019)	参与方	成员推断/属性推断	辅助数据	主动/被动	LFW/FaceScrub/PIPA/Yelp-health/Yelp-author/FourSquare/CSI
Nasr等人[113] (2019)	服务器/参与方	成员推断	—	主动/被动	CIFAR-100/Purchase/Texas100
Chen等人[114] (2020)	参与方	成员推断	—	被动	CIFAR-10/MNIST
Zhang等人[115] (2020)	参与方	成员推断	—	被动	MNIST
Hitaj等人[116] (2017)	参与方	数据重构	—	主动	AT&T/MNIST
Wang等人[117] (2019)	服务器	数据重构	辅助数据	主动/被动	AT&T/MNIST
Song等人[118] (2020)	服务器	数据重构	辅助数据	主动/被动	AT&T/MNIST
Zhu等人[16] (2019)	服务器	数据重构	—	被动	BERT/CIFAR-100/LFW/MNIST/SVHN
Geiping等人[119] (2020)	服务器	数据重构	—	被动	CIFAR-10/ImageNet
Zhao等人[120] (2020)	服务器	数据重构	—	被动	CIFAR-100/LFW/MNIST
Wei等人[121] (2020)	服务器	数据重构	—	被动	CIFAR-10/CIFAR-100/LFW/MNIST
Shen等人[122] (2021)	服务器	属性推断	辅助数据	主动	CASIS-WebFace/CelebA/LFW/MNIST

注: 攻击目标中“属性推断”表示模型任务无关属性的推断攻击

3.3 隐私保护方法

针对联邦攻击面临的隐私风险, 许多学者研究了一系列隐私保护方法, 防止参与方隐私信息的泄露. 根据隐私保护采用的技术手段, 主要可分为以下 5 类: 安全多方计算、差分隐私、加密、混淆和共享部分参数, 此外还有个别文献提出一些其他方法.

3.3.1 安全多方计算

安全多方计算 (secure multi-party computation, SMC) 允许多个数据所有者在互不信任的情况下进行协同计算, 最早由 Yao 于 1982 年提出[137]. 联邦学习是由多个参与方和服务器合作训练全局模型, 可以引入安全多方计算保护参与方隐私.

SMC 的数学描述如下: 有 n 个参与方 $\{P_1, P_2, \ldots, P_n\}$, 并各自拥有秘密数据 $\{x_1, x_2, \ldots, x_n\}$, 他们共同计算一个约定函数 $f(x_1, x_2, \ldots, x_n)=(y_1, y_2, \ldots, y_n)$, 其中 y_i 为 P_i 获得的输出结果. 在计算过程中, P_i 除了 y_i 外无法获知其他参与方的输入数据, 即 $x_j\ (i\neq j)$. SMC 是密码学技术的综合运用, 可以通过函数加密、秘密共享等技术实现.

Xu 等人[138]利用函数加密 (functional encryption)[139]实现 SMC. 函数加密是一个公钥加密系统, 任何人可以用公钥加密明文 m 得到密文 $Enc(m)$, 私钥持有者可以向某个函数 f 颁发一个密钥 key, 其他人可以基于 key 和 $Enc(m)$ 计算 $f(m)$, 而且在计算过程中无法获得关于 m 的任何信息. 因此 Xu 等人[138]在联邦学习系统中引入一个可信的第三方 (trusted third party, TPA), 负责公私钥管理和颁发 key. 参与方在上传模型更新时用公钥进行加密, 服务器在确定本轮迭代聚合的参与方集合后通知 TPA, TPA 生成对应的 key 发送给服务器, 服务器利用 key 和参与方加密的模型更新聚合全局模型, 从而保证在聚合过程中服务器无法获取任意参与方真实的模型更新.

Khazbak 等人[60]则利用秘密共享 (secret sharing) 技术实现 SMC. 秘密共享是将秘密进行拆分, 交由不同的参与者进行管理, 需要多个参与者协作且合作数量超过阈值时才可以恢复秘密, 其形式化定义如下:

$$S(s,t,n) \to \{\langle s_0\rangle, \langle s_1\rangle, \ldots, \langle s_n\rangle\} \tag{6}$$

$$R(\langle s_0\rangle, \langle s_1\rangle, \ldots, \langle s_m\rangle) \to s,\ t \leqslant m \leqslant n \tag{7}$$

其中, $S()$ 是拆分函数, s 为要拆分的秘密, t 为恢复门限, n 为拆分数量, $R()$ 为恢复函数, m 为协作参与者的数量. Khazbak 等人[60]设计的联邦学习系统需要两台聚合服务器. 在每轮迭代中, 参与方利用秘密共享将本地模型更新拆分成两部分, 分别发给不同的服务器, 之后每个服务器先聚合自身拥有的参与方共享, 再联合起来计算全局模型. 在这过程中每台服务器都只能获取参与方的部分模型更新, 无法从中推断参与方的隐私信息. 这种方法要求服务器之间不能相互勾结, 可以通过扩展更多的聚合服务器增强隐私保护效果. 董业等人[140]则将秘密共享与 Top-K 梯度选择相结合, 设计一种既高效又可以保护参与方隐私的联邦学习方案, 实现隐私保护、通信开销和模型性能三者间的平衡.

Li 等人[141]则将参与方划分到不同的链, 每条链上的参与方将本地模型更新与上一个节点的更新聚合后, 发送给下一个节点, 链尾节点将整条链聚合后的更新发送给服务器, 最终服务器聚合所有链的更新. 在这过程中, 服务器只能感知每条链聚合的模型参数, 无法获取单个参与方的模型更新.

3.3.2 差分隐私

差分隐私 (differential privacy) 是一种广泛应用的隐私保护技术, 它通过在用户的数据上添加扰动, 保证在一定概率范围内, 攻击者无法从用户发布的信息中推导出用户的隐私. 差分隐私的具体定义如下[142]: 一个随机化算法 M 提供 ε-差分隐私保护, 当且仅当对于任意两个只相差一条数据的邻近数据集 D 和 D'满足以下公式:

$$\Pr[M(D) \in S_M] \leqslant \exp(\varepsilon) \times \Pr[M(D') \in S_M] \tag{8}$$

其中, Pr 为算法 M 的输出概率, $M(D)$ 和 $M(D')$ 为算法 M 在数据集 D 和 D'上的输出, S_M 为 M 值域的子集. ε 是隐私保护预算 (privacy budget), 代表隐私保护的标准. ε 值越小, 标准越严格, 输出概率越接近, 隐私保护效果越好.

基于差分隐私的特性, 可以将聚合算法作为 M, 通过在参与方的模型更新 D 上添加噪声成为 D', 使聚合的全局模型与真实的全局模型尽可能接近, 同时也可以防止攻击者从 D'中推断出参与方的隐私信息. 差分隐私也可以在全局模型上应用, 以保护模型隐私.

文献 [143,144] 通过在服务器聚合全局模型时添加噪声, 隐藏单个参与方对全局模型的贡献, 实现客户端级别的隐私保护. 其中 Geyer 等人[143]是通过添加高斯噪声实现差分隐私, 而 Jayaraman 等人[144]添加的是拉普拉斯噪声. 文献 [145–148] 则要求参与方上传模型更新时在本地添加噪声, 防止服务器推断参与方的隐私信息. 其中 Triastcyn 等人[147]利用联邦学习应用中同个地区的参与方拥有近似数据分布的特点, 采用贝叶斯差分隐私减少指定 ε 下需要添加的噪声. 而 Zhao 等人[148]针对联邦学习的物联网应用, 提出新的本地差分隐私方法, 提高模型在特定 ε 下的表现. 针对数据不平衡的联邦学习场景, Huang 等人[146]提出在模型训练过程中根据梯度下降方向动态调整添加的噪声, 提升模型性能. Wei 等人[149]则分析证明给定 ε 存在最优的训练迭代轮数, 他们设计一个动态调整迭代轮数的算法, 在满足本地差分隐私的同时提高模型的收敛表现. 而 Wu 等人[150]在参与方本地训练过程多次添加高斯噪声, 以满足差分隐私.

在联邦学习中应用差分隐私并不会额外增加过多的计算开销, 还可以与其他隐私保护方案相结合增强保护效果, 但是它不可避免地会降低模型的准确性, 因此需要在隐私预算和模型性能之间进行平衡.

3.3.3 加 密

加密是利用密码学算法将联邦学习的模型更新转换为密文进行计算, 避免隐私数据直接暴露在攻击者面前, 主要是利用同态加密 (homomorphic encryption, HE) 算法实现.

HE 是一种允许用户直接在密文上进行运算的加密方法, 运算结果仍是密文, 且解密后与直接在明文上运算的结果是一致的, 即满足以下公式[151]:

$$Dec(Enc(m_1)\odot Enc(m_2)) = m_1 \oplus m_2 \tag{9}$$

其中, $Dec()$ 和 $Enc()$ 分别是解密运算和加密运算, m_1 和 m_2 是明文, $\odot$ 和 $\oplus$ 分别是在密文域和明文域上的运算. 根据密文支持的运算和次数, HE 可以分为全同态加密、类同态加密和部分同态加密[151].

(1) 全同态加密 (fully homomorphic encryption, FHE): $\odot$ 和 $\oplus$ 支持任意运算, 且运算次数不限.

(2) 类同态加密 (somewhat homomorphic encryption, SHE): $\odot$ 和 $\oplus$ 同时支持加法运算和乘法运算, 但运算次数有限.

(3) 部分同态加密 (partially homomorphic encryption, PHE): $\odot$ 和 $\oplus$ 只支持加法运算或乘法运算, 可细分为加法同态加密 (additive homomorphic encryption, AHE) 和乘法同态加密 (multiplication homomorphic encryption, MHE).

表 6 从加密类型、加密算法、防御的攻击方和参与方是否共享密钥这 4 个维度对基于同态加密的联邦学习隐私保护方案进行对比.

表 6 基于同态加密的联邦学习隐私保护方案对比

文献	加密类型	加密算法	攻击方	参与方是否共享密钥
Phong等人[152] (2017)	AHE	LWE-based[153]	服务器	是
Hao等人[154] (2019)	AHE	PPDM[155]	服务器	是
Chai等人[156] (2020)	AHE	Paillier[157]	服务器	是
Fang等人[158] (2020)	MHE	Double-key ElGamal	服务器/参与方	否
Hao等人[159] (2020)	FHE+AHE	BGV[160]+A-LWE[161]	服务器/参与方	否
Fang等人[162] (2021)	MHE	ElGamal[163]	服务器/参与方	否
Froelicher等人[164] (2021)	FHE	multiparty lattice-based[165]	服务器/参与方	否
Sav等人[166] (2021)	FHE	multiparty lattice-based[165]	服务器/参与方	否

文献 [152,154,156,158,159,162] 都是利用同态加密算法对参与方上传的模型更新进行加密, 使服务器聚合更新密文, 防止服务器提取参与方隐私. 在文献 [152,154,156] 中, 所有参与方共享一对公私钥, 在上传更新时用公钥进行加密, 等服务器聚合后在本地用私钥解密全局模型, 但这种方案只适用于不存在恶意方的场景, 否则恶意方可以与服务器勾结还原其他参与方的梯度. 为解决上述问题, 文献 [158,159,162] 分别提出不同的改进方案, 其中文献 [158,159] 采用不同的算法实现二次加密: 参与方先使用各自的公钥加密更新, 再用共享的公钥二次加密. Fang 等人[162]则利用秘密共享技术拆分原先参与方共享的私钥并交由不同的参与方保存, 服务器需要与多个恶意方勾结才可以还原正常参与方的梯度, 从而增加攻击难度. 文献 [164,166] 为进一步保护隐私, 利用 FHE 使参与方在本地对加密模型进行训练, 整个联邦学习训练过程中不存在明文的模型梯度, 还可以提供后量子 (post-quantum) 安全.

Li 等人[119]则设计一种非同态的加密方法 NIT, 可以将参与方的模型更新转化为无信息的分布, 消除模型更新携带的多媒体特征. 服务器基于加密梯度进行聚合, 参与方通过解密还原出全局模型. NIT 只支持 FedAvg[3]和 FSVRG[167]两种聚合算法.

基于加密的隐私保护方案受限于加密算法, 目前只支持简单的聚合算法, 且同态加密会引入大量通信和计算开销.

3.3.4 混 淆

混淆 (masking) 是指对参与方的模型更新进行混淆, 使攻击者无法从中推断出参与方隐私, 同时又可以保证混淆后模型更新的聚合结果是正确的.

Bonawitz 等人[168]提出了一种简单的混淆方案: 假定任意两个参与方 (u, v) 事先协商了一个随机矢量 $s_{u,v}$, 如果 u 在模型更新中添加 $s_{u,v}$, 而 v 减去 $s_{u,v}$, 则 u 和 v 相加聚合时混淆会抵消, 而且过程中 u 和 v 真实的模型更新也不会透露. 参与方 u 的混淆更新的具体公式如下:

$$y_u = x_u + \sum_{v \in U: u<v} s_{u,v} - \sum_{v \in U: u>v} s_{u,v} \tag{10}$$

其中, x_u 是参与方 u 真实的模型更新, U 是所有参与方的集合, u 和 v 的大小关系可以通过比较参与方的 ID 确定. 显然混淆模型更新的聚合结果是正确的:

$$z = \sum_{u \in U} y_u = \sum_{u \in U} x_u \tag{11}$$

但是上述混淆方案只适用于聚合所有参与方的场景, 聚合过程中如果部分参与方掉线会导致错误的聚合结果且无法恢复. 对此 Bonawitz 等人[168]利用秘密共享技术, 要求参与方 u 将 $s_{u,v}$ 拆分后发送给其他参与方, 后续聚合时即使 u 掉线, 服务器也可以从在线参与方收集共享的秘密恢复 $s_{u,v}$. 但这也为服务器推导 x_u 创造了条件, 服务器可以基于 y_u 和借助其他参与方恢复的 $s_{u,v}$ 计算 x_u. 因此 Bonawitz 等人[168]最终提出一个双重混淆方案:

$$y_u = x_u + PRG(b_u) + \sum_{v \in U: u<v} PRG(s_{u,v}) - \sum_{v \in U: u>v} PRG(s_{u,v}) \tag{12}$$

其中, PRG 是伪随机生成器, $PRG(s_{u,v})$ 表示以 $s_{u,v}$ 为种子生成随机数, b_u 为参与方 u 选择的随机数. 参与方 u 会将 b_u 和 $s_{u,v}$ 通过秘密共享的方式发送给其他参与方, 并限制每个参与方只能单独响应服务器对 b_u 或 $s_{u,v}$ 共享的请求, 增加服务器同时恢复 b_u 和 $s_{u,v}$ 的难度, 从而保护参与方 u 的隐私. 在聚合过程中, 服务器需要恢复所有掉线参与方的 $s_{u,v}$ 和所有在线参与方的 b_u.

在联邦学习的推荐系统应用中, 参与方可以根据本地数据的索引从聚合服务器选择下载部分模型、训练并上传. 为避免服务器从参与方下载和上传的模型中推断出参与方的数据索引, Niu 等人[169]提出参与方在每轮迭代中生成随机的索引集合, 按新的索引集合下载模型并训练, 同时采用双重混淆方案上传模型更新, 从而迷惑服务器, 使服务器无法推断参与方真实的数据索引.

基于混淆的隐私保护方案主要用于防范不可信的服务器, 需要参与方之间相互通信协商. 对于存在恶意方的攻击场景, 需要借助第三方的公钥基础设施 (public key infrastructure) 保证参与方之间通信消息的准确性.

3.3.5 共享部分参数

为解决参与方上传的模型梯度泄露本地数据隐私的问题, 部分学者提出只上传梯度的部分参数, 减少梯度泄露的隐私. 这类方法的难点在于减少参与方上传参数的同时如何保证全局模型的性能. 文献 [138,170,171] 根据参数绝对值和参数对全局模型的影响成正相关的特点, 让每个参与方按照一定比例选择绝对值较大的梯度参数上传, 同时将其他参数置零. 文献评估了不同比例系数对全局模型性能的影响, 通过实验证明参与方只上传部分参数也可以聚合出高性能的全局模型. 对于带有批标准化 (batch normalization, BN)[172]层的联邦学习模型, Andreux 等人[173]提出参与方只上传 BN 层的学习参数, 而在本地保留 BN 层的统计信息, 从而减少隐私的泄露, 并提高联邦学习在数据异构场景中的表现.

虽然共享部分参数的计算开销低, 在部分场景中防御效果明显, 但是其具体可提供的隐私保护能力尚未得到充分验证.

3.3.6 其 他

除了上述 5 种主要的隐私保护技术手段外, 部分学者也提出一些其他方法应对联邦学习的隐私攻击, 其中包括: 在 Chang 等人[99]提出的联邦学习框架 Cronus 中, 参与方利用数据标签替代模型参数, 避免参与方上传模型梯

度, 减少隐私泄露. Zhu 等人[16]则针对他们提出的数据重构攻击设计一些防护手段, 通过实验证明梯度压缩和稀疏化、提高模型训练的批处理大小以及增加输入图片的分辨率都可以有效降低攻击的成功率. So 等人[174]则利用随机量子化 (stochastic quantization) 和秘密共享技术实现隐私保护: 首先参与方将量子化后的模型更新分割、共享给其他参与方, 由服务器选择本轮聚合的参与方集合并广播; 随后参与方对共享的秘密求和后上传到服务器, 服务器从中还原聚合后的模型更新, 从而避免服务器直接接触参与方的更新. Chamikara 等人[175]则是在参与方的训练数据添加扰动, 使参与方训练的模型更新与真实更新有所偏差, 攻击者无法从中推断真实的训练数据, 但是这种方法会降低全局模型的准确性. Zhao 等人[171]则在聚合服务器和参与方之间引入可信的代理服务器, 避免聚合服务器和参与方的直接通信, 实现参与方匿名化.

3.3.7 综合保护方法

上述隐私保护技术在应用中可以相互结合, 增强保护效果. 表 7 对现有的综合隐私保护方案采用的技术进行总结, 从中可以推断出差分隐私作为一种通用的辅助手段被广泛应用到其他隐私保护方法中.

表 7 联邦学习综合隐私保护方案总结

文献	函数加密	差分隐私	加密	混淆	共享部分参数
Xu等人 (2019)[138]	√	√	—	—	—
Fang等人 (2020)[158]	—	—	√	—	√
Hao等人 (2020)[159]	—	√	√	—	—
Niu等人 (2020)[169]	—	√	—	√	—
Li等人 (2019)[170]	—	√	—	—	√
Zhao等人 (2021)[171]	—	√	—	—	√

3.4 与集中学习的共性和差异

本节从威胁模型、隐私攻击手段和隐私保护方法这 3 个方面对集中学习和联邦学习在隐私问题上的共性和差异进行分析.

3.4.1 威胁模型的异同

表 8 对集中学习和联邦学习在隐私威胁模型的共性与差异进行总结. 从表 8 可以看出: 首先在集中学习和联邦学习中, 数据隐私都属于攻击者的目标, 都包括成员推断和属性推断. 其次, 在攻击者知识方面, 研究两者的隐私攻击时都会考虑攻击者是否拥有与训练数据相似的辅助数据集.

表 8 集中学习与联邦学习的隐私威胁模型的对比

威胁模型		集中学习	联邦学习
攻击者角色	服务器	—	√
	参与方	—	√
	第三方	√	√
攻击者目标	成员推断	√	√
	属性推断	√	√
	模型萃取	√	—
攻击者知识	辅助数据集	√	√
攻击者模型	主动攻击	—	√
	被动攻击	√	√

而在威胁模型的其他方面, 集中学习与联邦学习存在如下差异.

(1) 集中学习的攻击者都是第三方, 企图提取服务提供商的模型隐私或数据隐私, 而联邦学习系统包含多类角色, 因此学者会针对服务器、参与方和第三方等不同角色研究不同的攻击手段和防护方法.

(2) 在集中学习中, 除了训练数据, 模型也是攻击者的目标之一. 例如, 在目前流行的 MLaaS (machine learning

as a service) 平台上, 对外提供付费人工智能服务的模型也具有一定的商业价值, 因此攻击者会通过构造特定的输入, 根据模型的返回结果尝试逆向提取目标模型的结构和参数信息, 从而复制一个功能相似甚至相同的模型, 这种攻击称为模型萃取 (model extraction) 攻击[176,177]. 而联邦学习现有的隐私研究主要探讨来自系统内部的威胁, 模型信息对攻击者是可见的, 因此目前没有联邦学习模型萃取攻击的相关研究.

(3) 在攻击模式方面, 因为集中学习的模型通常是在服务器上训练, 攻击者难以干预模型训练, 所以只在模型的推理阶段实施攻击, 不存在攻击模式上的差异. 而联邦学习的恶意服务器或恶意方可以通过干预训练阶段以获取更多的隐私.

3.4.2 隐私攻击手段的异同

在隐私攻击手段方面, 集中学习与联邦学习的隐私攻击有明显差异, 需要分别进行研究.

目前集中学习面临的隐私风险都属于黑盒攻击, 因此集中学习的成员推断主要是利用模型在训练数据和测试数据上的输出差异进行区分, 而属性推断则是通过大量的查询结果提取模型输出与某些特定属性的关联实现. 虽然这些攻击手段都是针对单个模型, 可以应用于联邦学习的全局模型, 但是联邦学习的聚合操作会削弱每个参与方的本地数据对全局模型的影响, 导致攻击成功率较低[113]. 另外, 集中学习的模型萃取攻击也是一个研究热点.

而联邦学习也面临着特有的隐私风险: 首先, 联邦学习的模型信息和梯度对攻击者是可见的, 这些都可以用于提取参与方的数据隐私[109,110]; 其次, 联邦学习的多轮迭代会暴露更多数据信息[113]; 最后, 攻击者通过干扰模型训练过程可以诱导目标泄露更多信息. 这 3 个因素都导致联邦学习的隐私攻击有别于集中学习.

3.4.3 隐私保护方法的异同

在隐私保护方法上, 集中学习和联邦学习存在以下共同点: 目前, 集中学习针对隐私攻击的防御主要是使用差分隐私和同态加密[107], 通过在训练数据、损失函数、梯度和参数上添加噪声, 或者用加密数据进行训练等方式, 减少隐私泄露. 这些都可以直接应用于联邦学习的参与方, 提高隐私保护能力.

而两者在隐私保护方法上的差异具体如下.

(1) 与两者在安全防护方法的差异类似, 在联邦学习应用集中学习的隐私保护方法时, 需要在资源开销和隐私性之间进行平衡, 以适用于联邦学习的参与方资源有限的场景.

(2) 因为联邦学习除模型训练外还包含聚合操作, 所以在应用差分隐私和加密方法时, 还需要解决聚合操作的差分隐私保证和同态加密有效性的问题.

(3) 因为联邦学习需要参与方和服务器之间多方协作, 所以服务器和参与方通信数据的隐私也是需要重点关注的问题, 因此发展出安全多方计算、混淆和共享部分参数等隐私保护技术.

4 未来展望

虽然联邦学习模型的安全与隐私研究已经取得许多研究成果, 但是目前还处于初期探索阶段, 尚有诸多问题亟待解决, 其中有以下 3 个重要问题值得深入研究.

(1) 成本低和隐蔽性强的联邦学习投毒攻击与防护

目前联邦学习安全攻击的研究主要集中在模型投毒攻击, 攻击者通过构造恶意的模型更新破坏全局模型, 许多学者在此之上进行攻防博弈. 然而, 模型投毒要求攻击者完全控制单个或多个参与方, 随着联邦学习部署应用的延伸, 逐渐减少的脆弱参与方将限制模型投毒的应用. 与之相比, 数据投毒对攻击者能力要求低, 具有更广泛的实施场景, 且在大规模训练数据集中更不易被发现. 然而, 目前对数据投毒的研究还比较浅显, 只停留在简单验证攻击可行性的阶段. 数据投毒需要经过模型本地训练阶段, 其产生的恶意更新与正常更新有一定的相似性, 是否可以生成恶意训练数据模糊恶意更新与正常更新, 以绕过现有异常检测聚合算法的防御?是否可以通过构造恶意数据生成目标模型更新, 从而利用现有模型投毒的研究成果实施更加隐蔽的攻击?如何防止数据投毒的攻击效果被模型聚合削弱?这些问题都亟待后续深入研究. 加强对联邦学习数据投毒的研究, 可以对联邦学习的安全性有更加深刻的认识, 进而推动联邦学习安全防护方法的探索, 为联邦学习的推广应用保驾护航.

(2) 参与方退出联邦学习时的隐私保护

在 GDPR 等隐私保护的法律法规中明确规定个人对其隐私数据享有删除权和被遗忘权, 即个人有权要求数据控制者删除其个人信息, 且数据控制者需采取必要的措施, 负责消除已经扩散出去的个人数据[2]. 在联邦学习应用中, 当个体参与方退出联邦学习系统时, 服务器需要按照法律规定删除参与方的个人信息. 从隐私攻击方法的总结可以发现参与方的本地数据会在模型参数留下痕迹, 因此服务器需要从模型参数中“忘却 (unlearning)”参与方的本地数据. 集中学习也面临着相同的隐私保护问题, Bourtoule 等人[178]提出通过排除目标数据重新训练模型解决, 但在联邦学习中模型参数已经通过多轮迭代扩散到其他参与方, 清除其他参与方本地模型的隐私痕迹变得非常困难. 因此, 需要研究改进联邦学习的机制, 确保可以删除和遗忘退出参与方的隐私信息. 另外还需要考虑可证明性, 即服务器可以向参与方证明其个人信息及扩散的数据都已经清除.

(3) 安全和隐私并重的联邦学习系统

目前对于联邦学习安全和隐私的研究都是侧重单个方面, 但在实际应用中安全威胁和隐私风险是同时存在的, 且无法通过简单叠加现有的安全防护手段和隐私保护方法进行防御, 例如差分隐私添加的噪声可能干扰安全聚合算法的检测, 同态加密的密文可能屏蔽模型更新的差异使安全聚合算法失效. 因此需要综合考虑联邦学习的安全和隐私问题, 研究安全与隐私并重的联邦学习系统. 文献 [60,99,179] 对此进行了初步的探索, 但是只涵盖部分安全威胁和隐私风险, 还有待更加全面的研究.

5 结束语

随着联邦学习的快速发展和广泛应用, 联邦学习模型的安全和隐私问题吸引了许多学者的兴趣和关注, 产生了不少瞩目的研究成果, 但目前相关的研究还处于初级阶段, 尚有许多关键问题亟待解决. 本文在充分调研和深入分析的基础上, 对联邦学习在安全和隐私领域最新的研究成果进行综述, 系统总结了联邦学习存在的安全和隐私攻击, 并对现有的防护方法进行科学的分类和分析. 同时, 本文也指出了当前联邦学习在安全和隐私领域尚未解决的问题, 并探讨未来的研究方向.

References:

[1] Liu JX, Meng XF. Survey on privacy-preserving machine learning. Journal of Computer Research and Development, 2020, 57(2): 346–362 (in Chinese with English abstract). [doi: 10.7544/issn1000-1239.2020.20190455]

[2] Regulation. Regulation (EU) 2016/679 of the European parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). Official Journal of the European Union, 2016, 119: 1–88.

[3] McMahan B, Moore E, Ramage D, Hampson S, Arcas BAY. Communication-efficient learning of deep networks from decentralized data. In: Proc. of the 20th Int'l Conf. on Artificial Intelligence and Statistics. Fort Lauderdale: JMLR, 2017. 1273–1282.

[4] Google AI Blog. Federated learning: Collaborative machine learning without centralized training data. 2017. http://ai.googleblog.com/2017/04/federated-learning-collaborative.html

[5] Mowla NI, Tran NH, Doh I, Chae K. Federated learning-based cognitive detection of jamming attack in flying ad-hoc network. IEEE Access, 2020, 8: 4338–4350. [doi: 10.1109/ACCESS.2019.2962873]

[6] Yang WS, Zhang YH, Ye KJ, Li L, Xu CZ. FFD: A federated learning based method for credit card fraud detection. In: Proc. of the 8th Int'l Conf. on Big Data. San Diego: Springer, 2019. 18–32. [doi: 10.1007/978-3-030-23551-2_2]

[7] Duan R, Boland MR, Liu ZX, Liu Y, Chang HH, Xu H, Chu HT, Schmid CH, Forrest CB, Holmes JH, Schuemie MJ, Berlin JA, Moore JH, Chen Y. Learning from electronic health records across multiple sites: A communication-efficient and privacy-preserving distributed algorithm. Journal of the American Medical Informatics Association, 2020, 27(3): 376–385. [doi: 10.1093/jamia/ocz199]

[8] Li ZY, Roberts K, Jiang XQ, Long Q. Distributed learning from multiple EHR databases: Contextual embedding models for medical events. Journal of Biomedical Informatics, 2019, 92: 103138. [doi: 10.1016/j.jbi.2019.103138]

[9] Huang L, Shea AL, Qian HN, Masurkar A, Deng H, Liu DB. Patient clustering improves efficiency of federated machine learning to predict mortality and hospital stay time using distributed electronic medical records. Journal of Biomedical Informatics, 2019, 99: 103291. [doi: 10.1016/j.jbi.2019.103291]

[10] Brisimi TS, Chen RD, Mela T, Olshevsky A, Paschalidis IC, Shi W. Federated learning of predictive models from federated electronic health records. Int'l Journal of Medical Informatics, 2018, 112: 59–67. [doi: 10.1016/j.ijmedinf.2018.01.007]

[11] Kim YJ, Sun JM, Yu H, Jiang XQ. Federated tensor factorization for computational phenotyping. In: Proc. of the 23rd ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. Halifax: Association for Computing Machinery, 2017. 887–895. [doi: 10.1145/3097983.3098118]

[12] Huang QY, Li ZY, Xie WT, Zhang Q. Edge computing in smart homes. Journal of Computer Research and Development, 2020, 57(9): 1800–1809 (in Chinese with English abstract). [doi: 10.7544/issn1000-1239.2020.20200253]

[13] Yang Q, Liu Y, Chen TJ, Tong YX. Federated machine learning: Concept and applications. ACM Trans. on Intelligent Systems and Technology, 2019, 10(2): 12. [doi: 10.1145/3298981]

[14] Blanchard P, El Mhamdi EM, Guerraoui R, Stainer J. Machine learning with adversaries: Byzantine tolerant gradient descent. In: Proc. of the 31st Int'l Conf. on Neural Information Processing Systems. Long Beach: ACM, 2017. 118–128. [doi: 10.5555/3294771.3294783]

[15] Yin D, Chen YD, Ramchandran K, Bartlett P. Byzantine-robust distributed learning: Towards optimal statistical rates. In: Proc. of the 35th Int'l Conf. on Machine Learning. Stockholm: PMLR, 2018. 5650–5659.

[16] Zhu LG, Liu ZJ, Han S. Deep leakage from gradients. In: Proc. of the 33rd Int'l Conf. on Neural Information Processing Systems. Vancouver: NIPS, 2019. 14747–14756.

[17] Phong LT, Aono Y, Hayashi T, Wang LH, Moriai S. Privacy-preserving deep learning via additively homomorphic encryption. IEEE Trans. on Information Forensics and Security, 2018, 13(5): 1333–1345. [doi: 10.1109/TIFS.2017.2787987]

[18] Jere MS, Farnan T, Koushanfar F. A taxonomy of attacks on federated learning. IEEE Security & Privacy, 2021, 19(2): 20–28. [doi: 10.1109/MSEC.2020.3039941]

[19] Xue MF, Yuan CX, Wu HY, Zhang YS, Liu WQ. Machine learning security: Threats, countermeasures, and evaluations. IEEE Access, 2020, 8: 74720–74742. [doi: 10.1109/ACCESS.2020.2987435]

[20] Zhang SS, Zuo X, Liu JW. The problem of the adversarial examples in deep learning. Chinese Journal of Computers, 2019, 42(8): 1886–1904 (in Chinese with English abstract). [doi: 10.11897/SP.J.1016.2019.01886]

[21] Tolpegin V, Truex S, Gursoy ME, Liu L. Data poisoning attacks against federated learning systems. In: Proc. of the 25th European Symp. on Computer Security. Guildford: Springer, 2020. 480–501. [doi: 10.1007/978-3-030-58951-6_24]

[22] Zhang JL, Chen JJ, Wu D, Chen B, Yu S. Poisoning attack in federated learning using generative adversarial nets. In: Proc. of the 18th IEEE Int'l Conf. on Trust, Security and Privacy in Computing and Communications/the 13th IEEE Int'l Conf. on Big Data Science and Engineering (TrustCom/BigDataSE). Rotorua: IEEE, 2019. 374–380. [doi: 10.1109/TrustCom/BigDataSE.2019.00057]

[23] Zhang JL, Chen B, Cheng X, Binh HTT, Yu S. PoisonGAN: Generative poisoning attacks against federated learning in edge computing systems. IEEE Internet of Things Journal, 2021, 8(5): 3310–3322. [doi: 10.1109/JIOT.2020.3023126]

[24] El Mhamdi EM, Guerraoui R, Rouault S. The hidden vulnerability of distributed learning in Byzantium. In: Proc. of the 35th Int'l Conf. on Machine Learning. Stockholm: PMLR, 2018. 3521–3530.

[25] Baruch M, Baruch G, Goldberg Y. A little is enough: Circumventing defenses for distributed learning. In: Proc. of the 33rd Int'l Conf. on Neural Information Processing Systems. Red Hook: ACM, 2019. 775. [doi: 10.5555/3454287.3455062]

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

[27] Bhagoji AN, Chakraborty S, Mittal P, Calo S. Analyzing federated learning through an adversarial lens. In: Proc. of the 36th Int'l Conf. on Machine Learning. Long Beach: ICML, 2019. 1012–1021.

[28] Fang MH, Cao XY, Jia JY, Gong NZ. Local model poisoning attacks to Byzantine-robust federated learning. In: Proc. of the 29th USENIX Conf. on Security Symp. Berkeley: USENIX Association, 2020. 1623–1640. [doi: 10.5555/3489212.3489304]

[29] Shejwalkar V, Houmansadr A. Manipulating the byzantine: Optimizing model poisoning attacks and defenses for federated learning. In: Proc. of the 2021 Network and Distributed Systems Security (NDSS) Symp. NDSS, 2021. 18.

[30] Nuding F, Mayer R. Poisoning attacks in federated learning: An evaluation on traffic sign classification. In: Proc. of the 10th ACM Conf. on Data and Application Security and Privacy. New York: Association for Computing Machinery, 2020. 168–170. [doi: 10.1145/3374664.3379534]

[31] Nguyen TD, Rieger P, Miettinen M, Sadeghi AR. Poisoning attacks on federated learning-based iot intrusion detection system. In: Proc. of the 2020 Workshop Decentralized IoT Systems and Securit (DISS). San Diego: DISS, 2020. 1–7.

[32] Bagdasaryan E, Veit A, Hua YQ, Estrin D, Shmatikov V. How to backdoor federated learning. In: Proc. of the 23rd Int'l Conf. on Artificial Intelligence and Statistics. Palermo: PMLR, 2020. 2938–2948.

[33] Sun ZT, Kairouz P, Suresh AT, McMahan HB. Can you really backdoor federated learning? arXiv:1911.07963, 2019.

[34] Xie CL, Huang KL, Chen PY, Li B. DBA: Distributed backdoor attacks against federated learning. In: Proc. of the 8th Int'l Conf. on Learning Representations. Addis Ababa: ICLR, 2020.

[35] Biggio B, Nelson B, Laskov P. Poisoning attacks against support vector machines. In: Proc. of the 29th Int'l Conf. on Machine Learning. Edinburgh: ACM, 2012. 1467–1474. [doi: 10.5555/3042573.304276]

[36] Goodfellow IJ, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y. Generative adversarial nets. In: Proc. of the 27th Int'l Conf. on Neural Information Processing Systems. Montreal: ACM, 2014. 2672–2680. [doi: 10.5555/2969033.2969125]

[37] Li T, Sahu AK, Zaheer M, Sanjabi M, Talwalkar A, Smith V. Federated optimization in heterogeneous networks. In: Proc. of the 3rd MLSys Conf. Austin, 2020. 429–450.

[38] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In: Proc. of the 3rd Int'l Conf. on Learning Representations. San Diego: ICLR, 2015.

[39] Gu TY, Dolan-Gavitt B, Garg S. BadNets: Identifying vulnerabilities in the machine learning model supply chain. arXiv:1708.06733, 2019.

[40] Krizhevsky A. Learning multiple layers of features from tiny images. Technical Report, Toronto: University of Toronto, 2009.

[41] Lecun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proc. of the IEEE, 1998, 86(11): 2278–2324. [doi: 10.1109/5.726791]

[42] Xiao H, Rasul K, Vollgraf R. Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. arXiv:1708.07747, 2017.

[43] Samaria FS, Harter AC. Parameterisation of a stochastic model for human face identification. In: Proc. of the 1994 IEEE Workshop on Applications of Computer Vision. Sarasota: IEEE, 1994. 138–142. [doi: 10.1109/ACV.1994.341300]

[44] Caldas S, Duddu SMK, Wu P, Li T, Konečný J, McMahan HB, Smith V, Talwalkar A. LEAF: A benchmark for federated settings. arXiv:1812.01097, 2019.

[45] Dua D, Graff C. UCI machine learning repository. 2017. https://archive.ics.uci.edu/ml/index.php

[46] Kather JN, Weis CA, Bianconi F, Melchers SM, Schad LR, Gaiser T, Marx A, Zöllner FG. Multi-class texture analysis in colorectal cancer histology. Scientific Reports, 2016, 6: 27988. [doi: 10.1038/srep27988]

[47] Serna CG, Ruichek Y. Classification of traffic signs: The European dataset. IEEE Access, 2018, 6: 78136–78148. [doi: 10.1109/ACCESS.2018.2884826]

[48] Cohen G, Afshar S, Tapson J, Van Schaik A. EMNIST: Extending mnist to handwritten letters. In: Proc. of the 2017 Int'l Joint Conf. on Neural Networks (IJCNN). Anchorage: IEEE, 2017. 2921–2926. [doi: 10.1109/IJCNN.2017.7966217]

[49] Tiny imagenet. 2021. https://kaggle.com/c/tiny-imagenet

[50] Nguyen TD, Marchal S, Miettinen M, Fereidooni H, Asokan N, Sadeghi AR. DÏoT: A federated self-learning anomaly detection system for IoI. In: Proc. of the 39th IEEE Int'l Conf. on Distributed Computing Systems (ICDCS). Dallas: IEEE, 2019. 756–767. [doi: 10.1109/ICDCS.2019.00080]

[51] Sivanathan A, Gharakheili HH, Loi F, Radford A, Wijenayake C, Vishwanath A, Sivaraman V. Classifying IoT devices in smart environments using network traffic characteristics. IEEE Trans. on Mobile Computing, 2019, 18(8): 1745–1759. [doi: 10.1109/TMC.2018.2866249]

[52] Acquire valued shoppers challenge. 2021. https://kaggle.com/c/acquire-valued-shoppers-challenge

[53] Loan data set. 2021. https://kaggle.com/burak3ergun/loan-data-set

[54] Chen YD, Su LL, Xu JM. Distributed statistical machine learning in adversarial settings: Byzantine gradient descent. Proc. of the 2017 ACM on Measurement and Analysis of Computing Systems, 2017, 1(2): 44. [doi: 10.1145/3154503]

[55] Xie C, Koyejo O, Gupta I. Generalized byzantine-tolerant SGD. arXiv:1802.10116, 2018.

[56] Cao D, Chang S, Lin ZJ, Liu GH, Sun DH. Understanding distributed poisoning attack in federated learning. In: Proc. of the 25th IEEE Int'l Conf. on Parallel and Distributed Systems (ICPADS). Tianjin: IEEE, 2019. 233–239. [doi: 10.1109/ICPADS47876.2019.00042]

[57] Lu YY, Fan L. An efficient and robust aggregation algorithm for learning federated CNN. In: Proc. of the 3rd Int'l Conf. on Signal Processing and Machine Learning. Beijing: Association for Computing Machinery, 2020. 1–7. [doi: 10.1145/3432291.3432303]

[58] Wu ZX, Ling Q, Chen TY, Giannakis GB. Federated variance-reduced stochastic gradient descent with robustness to Byzantine attacks. IEEE Trans. on Signal Processing, 2020, 68: 4583–4596. [doi: 10.1109/TSP.2020.3012952]

[59] Muñoz-González L, Co KT, Lupu EC. Byzantine-robust federated machine learning through adaptive model averaging. arXiv:1909.05125, 2019.

[60] Khazbak Y, Tan TX, Cao GH. MLGuard: Mitigating poisoning attacks in privacy preserving distributed collaborative learning. In: Proc.

of the 29th Int'l Conf. on Computer Communications and Networks (ICCCN). Honolulu: IEEE, 2020. 1–9. [doi: 10.1109/ICCCN49398.2020.9209670]

[61] Fung C, Yoon CJM, Beschastnikh I. The limitations of federated learning in sybil settings. In: Proc. of the 23rd Int'l Symp. on Research in Attacks, Intrusions and Defenses (RAID 2020). San Sebastian: USENIX Association, 2020. 301–316.

[62] Yang HB, Zhang X, Fang MH, Liu J. Byzantine-resilient stochastic gradient descent for distributed learning: A lipschitz-inspired coordinate-wise median approach. In: Proc. of the 58th IEEE Conf. on Decision and Control (CDC). Nice: IEEE, 2019. 5832–5837. [doi: 10.1109/CDC40024.2019.9029245]

[63] Yu L, Wu LF. Towards Byzantine-resilient federated learning via group-wise robust aggregation. In: Yang Q, Fan LX, Yu H, eds. Federated Learning: Privacy and Incentive. Cham: Springer, 2020. 81–92. [doi: 10.1007/978-3-030-63076-8_6]

[64] Singh AK, Blanco-Justicia A, Domingo-Ferrer J, Sánchez D, Rebollo-Monedero D. Fair detection of poisoning attacks in federated learning. In: Proc. of the 32nd IEEE Int'l Conf. on Tools with Artificial Intelligence (ICTAI). Baltimore: IEEE, 2020. 224–229. [doi: 10.1109/ICTAI50040.2020.00044]

[65] He L, Karimireddy SP, Jaggi M. Byzantine-robust learning on heterogeneous datasets via resampling. arXiv:2006.09365, 2020.

[66] Wang Y, Zhu TQ, Chang WH, Shen S, Ren W. Model poisoning defense on federated learning: A validation based approach. In: Proc. of the 14th Int'l Conf. on Network and System Security. Melbourne: Springer, 2020. 207–223. [doi: 10.1007/978-3-030-65745-1_12]

[67] Tan JJ, Liang YC, Luong NC, Niyato D. Toward smart security enhancement of federated learning networks. IEEE Network, 2021, 35(1): 340–347. [doi: 10.1109/MNET.011.2000379]

[68] Chen ZY, Tian P, Liao WX, Yu W. Zero knowledge clustering based adversarial mitigation in heterogeneous federated learning. IEEE Trans. on Network Science and Engineering, 2021, 8(2): 1070–1083. [doi: 10.1109/TNSE.2020.3002796]

[69] Xie C, Koyejo S, Gupta I. Zeno: Distributed stochastic gradient descent with suspicion-based fault-tolerance. In: Proc. of the 36th Int'l Conf. on Machine Learning. Long Beach: PMLR, 2019. 6893–6901.

[70] Cao XY, Fang MH, Liu J, Gong NZ. FLTrust: Byzantine-robust federated learning via trust bootstrapping. In: Proc. of the 2021 Network and Distributed Systems Security (NDSS) Symp. NDSS, 2021.

[71] Zhao Y, Chen JJ, Zhang JL, Wu D, Teng J, Yu S. PDGAN: A novel poisoning defense method in federated learning using generative adversarial network. In: Proc. of the 19th Int'l Conf. on Algorithms and Architectures for Parallel Processing. Australia: Springer, 2020. 595–609. [doi: 10.1007/978-3-030-38991-8_39]

[72] Zhao LC, Hu SS, Wang Q, Jiang JL, Shen C, Luo XY, Hu PF. Shielding collaborative learning: Mitigating poisoning attacks through client-side detection. IEEE Trans. on Dependable and Secure Computing, 2021, 18(5): 2029–2041. [doi: 10.1109/TDSC.2020.2986205]

[73] Bao XL, Su C, Xiong Y, Huang WC, Hu YF. FLChain: A blockchain for auditable federated learning with trust and incentive. In: Proc. of the 5th Int'l Conf. on Big Data Computing and Communications (BIGCOM). Qingdao: IEEE, 2019. 151–159. [doi: 10.1109/BIGCOM.2019.00030]

[74] Li YZ, Chen C, Liu N, Huang HW, Zheng ZB, Yan Q. A blockchain-based decentralized federated learning framework with committee consensus. IEEE Network, 2021, 35(1): 234–241. [doi: 10.1109/MNET.011.2000263]

[75] Shayan M, Fung C, Yoon CJM, Beschastnikh I. Biscotti: A blockchain system for private and secure federated learning. IEEE Trans. on Parallel and Distributed Systems, 2021, 32(7): 1513–1525. [doi: 10.1109/TPDS.2020.3044223]

[76] Zhao Y, Zhao J, Jiang LS, Tan R, Niyato D, Li ZX, Lyu LJ, Liu YB. Privacy-preserving blockchain-based federated learning for IoT devices. IEEE Internet of Things Journal, 2021, 8(3): 1817–1829. [doi: 10.1109/JIOT.2020.3017377]

[77] Liu Y, Peng JL, Kang JW, Iliyasu AM, Niyato D, El-Latif AAA. A secure federated learning framework for 5G networks. IEEE Wireless Communications, 2020, 27(4): 24–31. [doi: 10.1109/MWC.01.1900525]

[78] Zhao Y, Xu K, Wang HY, Li B, Jia RX. Stability-based analysis and defense against backdoor attacks on edge computing services. IEEE Network, 2021, 35(1): 163–169. [doi: 10.1109/MNET.011.2000265]

[79] Zhang JL, Wu D, Liu CY, Chen B. Defending poisoning attacks in federated learning via adversarial training method. In: Proc. of the 3rd Int'l Conf. on Frontiers in Cyber Security. Tianjin: Springer, 2020. 83–94. [doi: 10.1007/978-981-15-9739-8_7]

[80] Chen Y, Luo F, Li T, Xiang T, Liu ZL, Li J. A training-integrity privacy-preserving federated learning scheme with trusted execution environment. Information Sciences, 2020, 522: 69–79. [doi: 10.1016/j.ins.2020.02.037]

[81] Zhang XL, Li FT, Zhang ZY, Li Q, Wang C, Wu JP. Enabling execution assurance of federated learning at untrusted participants. In: Proc. of the 2020 IEEE Conf. on Computer Communications. Toronto: IEEE, 2020. 1877–1886. [doi: 10.1109/INFOCOM41043.2020.9155414]

[82] Peng Z, Xu JL, Chu XW, Gao S, Yao Y, Gu R, Tang YZ. VFChain: Enabling verifiable and auditable federated learning via blockchain systems. IEEE Trans. on Network Science and Engineering, 2022, 9(1): 173–186. [doi: 10.1109/TNSE.2021.3050781]

[83] Qu YY, Pokhrel SR, Garg S, Gao LX, Xiang Y. A blockchained federated learning framework for cognitive computing in industry 4.0 networks. IEEE Trans. on Industrial Informatics, 2021, 17(4): 2964–2973. [doi: 10.1109/TII.2020.3007817]

[84] Calauzènes C, Le Roux N. Distributed saga: Maintaining linear convergence rate with limited communication. arXiv:1705.10405, 2017.

[85] Diakonikolas I, Kamath G, Kane D, Li J, Steinhardt J, Stewart A. Sever: A robust meta-algorithm for stochastic optimization. In: Proc. of the 36th Int'l Conf. on Machine Learning. Long Beach: ICML, 2019. 1596–1606.

[86] Jain AK. Data clustering: 50 years beyond K-means. Pattern Recognition Letters, 2010, 31(8): 651–666. [doi: 10.1016/j.patrec.2009.09.011]

[87] Barreno M, Nelson B, Joseph AD, Tygar JD. The security of machine learning. Machine Learning, 2010, 81(2): 121–148. [doi: 10.1007/s10994-010-5188-5]

[88] Jagielski M, Oprea A, Biggio B, Liu C, Nita-Rotaru C, Li B. Manipulating machine learning: Poisoning attacks and countermeasures for regression learning. In: Proc. of the 2018 IEEE Symp. on Security and Privacy (SP). San Francisco: IEEE, 2018. 19–35. [doi: 10.1109/SP.2018.00057]

[89] Kate A, Zaverucha GM, Goldberg I. Constant-size commitments to polynomials and their applications. In: Proc. of the 16th Int'l Conf. on Advances in Cryptology. Singapore: Springer, 2010. 177–194. [doi: 10.1007/978-3-642-17373-8_11]

[90] Jakobsson M, Juels A. Proofs of work and bread pudding protocols(extended abstract). In: Preneel B, ed. Secure Information Networks. New York: Springer, 1999. 258–272. [doi: 10.1007/978-0-387-35568-9_18]

[91] Gilad Y, Hemo R, Micali S, Vlachos G, Zeldovich N. Algorand: Scaling byzantine agreements for cryptocurrencies. In: Proc. of the 26th Symp. on Operating Systems Principles. Shanghai: Association for Computing Machinery, 2017. 51–68. [doi: 10.1145/3132747.3132757]

[92] Bentov I, Lee C, Mizrahi A, Rosenfeld M. Proof of activity: Extending bitcoin's proof of work via proof of stake. ACM SIGMETRICS Performance Evaluation Review, 2014, 42(3): 34–37. [doi: 10.1145/2695533.2695545]

[93] McKeen F, Alexandrovich I, Berenzon A, Rozas CV, Shafi H, Shanbhogue V, Savagaonkar UR. Innovative instructions and software model for isolated execution. In: Proc. of the 2nd Int'l Workshop on Hardware and Architectural Support for Security and Privacy. Tel-Aviv: Association for Computing Machinery, 2013. 10. [doi: 10.1145/2487726.2488368]

[94] Wang J, Fan CY, Cheng YQ, Zhao B, Wei T, Yan F, Zhang HG, Ma J. Analysis and research on SGX technology. Ruan Jian Xue Bao/Journal of Software, 2018, 29(9): 2778–2798 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5594.htm [doi: 10.13328/j.cnki.jos.005594]

[95] Costan V, Devadas S. Intel SGX explained. In: Surhone LM, Tennoe MT, Henssonow SF, eds. Cryptology ePrint Archive. Whitefish: Betascript Publishing, 2016.

[96] Louppe G, Kagan M, Cranmer K. Learning to pivot with adversarial networks. In: Proc. of the 31st Int'l Conf. on Neural Information Processing Systems. Long Beach: ACM, 2017. 982–991. [doi: 10.5555/3294771.3294865]

[97] Ibitoye O, Shafiq MO, Matrawy A. DiPSeN: Differentially private self-normalizing neural networks for adversarial robustness in federated learning. arXiv:2101.03218, 2021.

[98] Goodfellow IJ, Shlens J, Szegedy C. Explaining and harnessing adversarial examples. In: Proc. of the 3rd Int'l Conf. on Learning Representations. San Diego: ICLR, 2015.

[99] Chang HY, Shejwalkar V, Shokri R, Houmansadr A. Cronus: Robust and heterogeneous collaborative learning with black-box knowledge transfer. arXiv:1912.11279, 2019.

[100] Kang JW, Xiong ZH, Niyato D, Zou YZ, Zhang Y, Guizani M. Reliable federated learning for mobile networks. IEEE Wireless Communications, 2020, 27(2): 72–80. [doi: 10.1109/MWC.001.1900119]

[101] Guo XJ, Liu ZL, Li J, Gao JQ, Hou BY, Dong CY, Baker T. VeriFL: Communication-efficient and fast verifiable aggregation for federated learning. IEEE Trans. on Information Forensics and Security, 2021, 16: 1736–1751. [doi: 10.1109/TIFS.2020.3043139]

[102] Li XJ, Wu GW, Yao L, Zhang WZ, Zhang B. Progress and future challenges of security attacks and defense mechanisms in machine learning. Ruan Jian Xue Bao/Journal of Software, 2021, 32(2): 406−423 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/6147.htm [doi: 10.13328/j.cnki.jos.006147]

[103] Dumford J, Scheirer W. Backdooring convolutional neural networks via targeted weight perturbations. In: Proc. of the 2020 IEEE Int'l Joint Conf. on Biometrics. Houston: IEEE, 2018. 1−9. [doi: 10.1109/IJCB48548.2020.9304875]

[104] Guo C, Wu RH, Weinberger KQ. On hiding neural networks inside neural networks. arXiv:2002.10078, 2021.

[105] Tang RX, Du MN, Liu NH, Yang F, Hu X. An embarrassingly simple approach for trojan attack in deep neural networks. In: Proc. of the 26th ACM SIGKDD Int'l Conf. on Knowledge Discovery & Data Mining. Association for Computing Machinery, 2020. 218–228. [doi: 10.1145/3394486.3403064]

[106] Li YM, Jiang Y, Li ZF, Xia ST. Backdoor learning: A survey. arXiv:2007.08745, 2022.

[107] Ji SL, Du TY, Li JF, Shen C, Li B. Security and privacy of machine learning models: A survey. Ruan Jian Xue Bao/Journal of Software, 2021, 32(1): 41–67 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/6131.htm [doi: 10.13328/j.cnki.jos.006131]

[108] Tan ZW, Zhang LF. Survey on privacy preserving techniques for machine learning. Ruan Jian Xue Bao/Journal of Software, 2020, 31(7): 2127–2156 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/6052.htm [doi: 10.13328/j.cnki.jos.006052]

[109] Shokri R, Stronati M, Song CZ, Shmatikov V. Membership inference attacks against machine learning models. In: Proc. of the 2017 IEEE Symp. on Security and Privacy. San Jose: IEEE, 2017. 3–18. [doi: 10.1109/SP.2017.41]

[110] Song CZ, Ristenpart T, Shmatikov V. Machine learning models that remember too much. In: Proc. of the 2017 ACM SIGSAC Conf. on Computer and Communications Security. Dallas: Association for Computing Machinery, 2017. 587–601. [doi: 10.1145/3133956.3134077]

[111] Fredrikson M, Jha S, Ristenpart T. Model inversion attacks that exploit confidence information and basic countermeasures. In: Proc. of the 22nd ACM SIGSAC Conf. on Computer and Communications Security. Denver: Association for Computing Machinery, 2015. 1322–1333. [doi: 10.1145/2810103.2813677]

[112] Melis L, Song CZ, De Cristofaro E, Shmatikov V. Exploiting unintended feature leakage in collaborative learning. In: Proc. of the 2019 IEEE Symp. on Security and Privacy. San Francisco: IEEE, 2019. 691–706. [doi: 10.1109/SP.2019.00029]

[113] Nasr M, Shokri R, Houmansadr A. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In: Proc. of the 2019 IEEE Symp. on Security and Privacy. San Francisco: 2019. 739–753. [doi: 10.1109/SP.2019.00065]

[114] Chen JL, Zhang JL, Zhao YC, Han H, Zhu K, Chen B. Beyond model-level membership privacy leakage: An adversarial approach in federated learning. In: Proc. of the 29th Int'l Conf. on Computer Communications and Networks. Honolulu: IEEE, 2020. 1–9. [doi: 10.1109/ICCCN49398.2020.9209744]

[115] Zhang JW, Zhang JL, Chen JJ, Yu S. GAN enhanced membership inference: A passive local attack in federated learning. In: Proc. of the 2020 IEEE Int'l Conf. on Communications. Dublin: IEEE, 2020. 1–6. [doi: 10.1109/ICC40277.2020.9148790]

[116] Hitaj B, Ateniese G, Perez-Cruz F. Deep models under the GAN: Information leakage from collaborative deep learning. In: Proc. of the 2017 ACM SIGSAC Conf. on Computer and Communications Security. Dallas: Association for Computing Machinery, 2017. 603–618. [doi: 10.1145/3133956.3134012]

[117] Wang ZB, Song MK, Zhang ZF, Song Y, Wang Q, Qi HR. Beyond inferring class representatives: User-level privacy leakage from federated learning. In: Proc. of the 2019 IEEE Conf. on Computer Communications. Paris: IEEE, 2019. 2512–2520. [doi: 10.1109/INFOCOM.2019.8737416]

[118] Song MK, Wang ZB, Zhang ZF, Song Y, Wang Q, Ren J, Qi HR. Analyzing user-level privacy attack against federated learning. IEEE Journal on Selected Areas in Communications, 2020, 38(10): 2430–2444. [doi: 10.1109/JSAC.2020.3000372]

[119] Geiping J, Bauermeister H, Dröge H, Moeller M. Inverting gradients—How easy is it to break privacy in federated learning? In: Proc. of the 34th Int'l Conf. on Neural Information Processing Systems. Vancouver: ACM, 2020. 1421. [doi: 10.5555/3495724.3497145]

[120] Zhao B, Mopuri KR, Bilen H. iDLG: Improved deep leakage from gradients. arXiv:2001.02610, 2020.

[121] Wei WQ, Liu L, Loper M, Chow KH, Gursoy ME, Truex S, Wu YZ. A framework for evaluating client privacy leakages in federated learning. In: Proc. of the 25th European Symp. on Computer Security. Guildford: Springer, 2020. 545–566. [doi: 10.1007/978-3-030-58951-6_27]

[122] Shen M, Wang H, Zhang B, Zhu LH, Xu K, Li Q, Du XJ. Exploiting unintended property leakage in blockchain-assisted federated learning for intelligent edge computing. IEEE Internet of Things Journal, 2021, 8(4): 2265–2275. [doi: 10.1109/JIOT.2020.3028110]

[123] Kingma DP, Ba J. Adam: A method for stochastic optimization. In: Proc. of the 3rd Int'l Conf. on Learning Representations. San Diego: ICLR, 2015.

[124] Zhang Y, Liu JW, Zuo X. Survey of multi-task learning. Chinese Journal of Computers, 2020, 43(7): 1340–1378 (in Chinese with English abstract). [doi: 10.11897/SP.J.1016.2020.01340]

[125] Huang GB, Mattar M, Berg T, Learned-Miller E. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. In: Proc. of the 2008 Workshop on Faces in 'Real-Life' Images: Detection, Alignment, and Recognition. Marseille: HAL, 2008.

[126] Ng HW, Winkler S. A data-driven approach to cleaning large face datasets. In: Proc. of the 2014 IEEE Int'l Conf. on Image Processing. Paris: IEEE, 2014. 343–347. [doi: 10.1109/ICIP.2014.7025068]

[127] Zhang N, Paluri M, Taigman Y, Fergus R, Bourdev L. Beyond frontal faces: Improving person recognition using multiple cues. In: Proc. of the 2015 IEEE Conf. on Computer Vision and Pattern Recognition. Boston: IEEE, 2015. 4804–4813. [doi: 10.1109/CVPR.2015.

7299113]

[128] Devlin J, Chang MW, Lee K, Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proc. of the 2019 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Minneapolis: Association for Computational Linguistics, 2019. 4171–4186. [doi: 10.18653/v1/N19-1423]

[129] Netzer Y, Wang T, Coates A, Bissacco A, Wu B, Ng AY. Reading digits in natural images with unsupervised feature learning. In: Proc. of the 2011 NIPS Workshop on Deep Learning and Unsupervised Feature Learning. NIPS, 2011.

[130] Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang ZH, Karpathy A, Khosla A, Bernstein M, Berg AC, Fei-Fei L. ImageNet large scale visual recognition challenge. Int'l Journal of Computer Vision, 2015, 115(3): 211–252. [doi: 10.1007/s11263-015-0816-y]

[131] Li SZ, Yi D, Lei Z, Liao SC. The CASIA NIR-VIS 2.0 face database. In: Proc. of the 2013 IEEE Conf. on Computer Vision and Pattern Recognition Workshops. Portland: IEEE, 2013. 348–353. [doi: 10.1109/CVPRW.2013.59]

[132] Liu ZW, Luo P, Wang XG, Tang XO. Deep learning face attributes in the wild. In: Proc. of the 2015 IEEE Int'l Conf. on Computer Vision. Santiago: IEEE, 2015. 3730–3738. [doi: 10.1109/ICCV.2015.425]

[133] Yelp dataset. 2021. https://www.yelp.com/dataset

[134] Yang DQ, Zhang DQ, Chen LB, Qu BQ. NationTelescope: Monitoring and visualizing large-scale collective behavior in LBSNs. Journal of Network and Computer Applications, 2015, 55: 170–180. [doi: 10.1016/j.jnca.2015.05.010]

[135] Texas Department of State Health Services. Hospital discharge data public use data file. 2021. https://www.dshs.texas.gov/THCIC/Hospitals/Download.shtm

[136] Verhoeven B, Daelemans W. CLiPS stylometry investigation (CSI) corpus: A dutch corpus for the detection of age, gender, personality, sentiment and deception in text. In: Proc. of the 9th Int'l Conf. on Language Resources and Evaluation. Reykjavik: European Language Resources Association, 2014. 3081–3085.

[137] Yao AC. Protocols for secure computations. In: Proc. of the 23rd Annual Symp. on Foundations of Computer Science. Chicago: IEEE, 1982. 160–164. [doi: 10.1109/SFCS.1982.38]

[138] Xu RH, Baracaldo N, Zhou Y, Anwar A, Ludwig H. HybridAlpha: An efficient approach for privacy-preserving federated learning. In: Proc. of the 12th ACM Workshop on Artificial Intelligence and Security. London: Association for Computing Machinery, 2019. 13–23. [doi: 10.1145/3338501.3357371]

[139] Boneh D, Sahai A, Waters B. Functional encryption: Definitions and challenges. In: Proc. of the 8th Theory of Cryptography Conf. Providence: Springer, 2011. 253–273. [doi: 10.1007/978-3-642-19571-6_16]

[140] Dong Y, Hou W, Chen XJ, Zeng S. Efficient and secure federated learning based on secret sharing and gradients selection. Journal of Computer Research and Development, 2020, 57(10): 2241–2250 (in Chinese with English abstract). [doi: 10.7544/issn1000-1239.2020.20200463]

[141] Li Y, Zhou YP, Jolfaei A, Yu DJ, Xu GC, Zheng X. Privacy-preserving federated learning framework based on chained secure multiparty computing. IEEE Internet of Things Journal, 2021, 8(8): 6178–6186. [doi: 10.1109/JIOT.2020.3022911]

[142] Xiong P, Zhu TQ, Wang XF. A survey on differential privacy and applications. Chinese Journal of Computers, 2014, 37(1): 101–122 (in Chinese with English abstract). [doi: 10.3724/SP.J.1016.2014.00101]

[143] Geyer RC, Klein T, Nabi M. Differentially private federated learning: A client level perspective. arXiv:1712.07557, 2018.

[144] Jayaraman B, Wang LX, Evans D, Gu QQ. Distributed learning without distress: Privacy-preserving empirical risk minimization. In: Proc. of the 32nd Int'l Conf. on Neural Information Processing Systems. Montréal: ACM, 2018. 6346–6357. [doi: 10.5555/3327345.3327531]

[145] Bhowmick A, Duchi J, Freudiger J, Kapoor G, Rogers R. Protection against reconstruction and its applications in private federated learning. arXiv:1812.00984, 2019.

[146] Huang XX, Ding Y, Jiang ZL, Qi SH, Wang X, Liao Q. DP-FL: A novel differentially private federated learning framework for the unbalanced data. World Wide Web, 2020, 23(4): 2529–2545. [doi: 10.1007/s11280-020-00780-4]

[147] Triastcyn A, Faltings B. Federated learning with Bayesian differential privacy. In: Proc. of the 2019 IEEE Int'l Conf. on Big Data. Los Angeles: IEEE, 2019. 2587–2596. [doi: 10.1109/BigData47090.2019.9005465]

[148] Zhao Y, Zhao J, Yang MM, Wang T, Wang N, Lyu LJ, Niyato D, Lam KY. Local differential privacy-based federated learning for internet of things. IEEE Internet of Things Journal, 2021, 8(11): 8836–8853. [doi: 10.1109/JIOT.2020.3037194]

[149] Wei K, Li J, Ding M, Ma C, Su H, Zhang B, Poor HV. User-level privacy-preserving federated learning: Analysis and performance optimization. IEEE Trans. on Mobile Computing, 2021. [doi: 10.1109/TMC.2021.3056991]

[150] Wu MQ, Ye DD, Ding JH, Guo YX, Yu R, Pan M. Incentivizing differentially private federated learning: A multidimensional contract

approach. IEEE Internet of Things Journal, 2021, 8(13): 10639–10651. [doi: 10.1109/JIOT.2021.3050163]

[151] Li ZY, Gui XL, Gu YJ, Li XS, Dai HJ, Zhang XJ. Survey on homomorphic encryption algorithm and its application in the privacy-preserving for cloud computing. Ruan Jian Xue Bao/Journal of Software, 2018, 29(7): 1827–1851 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5354.htm [doi: 10.13328/j.cnki.jos.005354]

[152] Phong LT, Aono Y, Hayashi T, Wang LH, Moriai S. Privacy-preserving deep learning: Revisited and enhanced. In: Proc. of the 8th Int'l Conf. on Applications and Techniques in Information Security. Auckland: Springer, 2017. 100–110. [doi: 10.1007/978-981-10-5421-1_9]

[153] Aono Y, Hayashi T, Phong LT, Wang LH. Efficient key-rotatable and security-updatable homomorphic encryption. In: Proc. of the 5th ACM Int'l Workshop on Security in Cloud Computing. Abu Dhabi: Association for Computing Machinery, 2017. 35–42. [doi: 10.1145/3055259.3055260]

[154] Hao M, Li HW, Xu GW, Liu S, Yang HM. Towards efficient and privacy-preserving federated deep learning. In: Proc. of the 2019 IEEE Int'l Conf. on Communications. Shanghai: IEEE, 2019. 1–6. [doi: 10.1109/ICC.2019.8761267]

[155] Zhou J, Cao ZF, Dong XL, Lin XD. PPDM: A privacy-preserving protocol for cloud-assisted e-healthcare systems. IEEE Journal of Selected Topics in Signal Processing, 2015, 9(7): 1332–1344. [doi: 10.1109/JSTSP.2015.2427113]

[156] Chai D, Wang LY, Chen K, Yang Q. Secure federated matrix factorization. IEEE Intelligent Systems, 2021, 36(5): 11–20. [doi: 10.1109/MIS.2020.3014880]

[157] Paillier P. Public-key cryptosystems based on composite degree residuosity classes. In: Proc. of the 1999 Int'l Conf. on the Theory and Application of Cryptographic Techniques. Prague: Springer, 1999. 223–238. [doi: 10.1007/3-540-48910-X_16]

[158] Fang C, Guo YB, Wang N, Ju AK. Highly efficient federated learning with strong privacy preservation in cloud computing. Computers & Security, 2020, 96: 101889. [doi: 10.1016/j.cose.2020.101889]

[159] Hao M, Li HW, Luo XZ, Xu GW, Yang HM, Liu S. Efficient and privacy-enhanced federated learning for industrial artificial intelligence. IEEE Trans. on Industrial Informatics, 2020, 16(10): 6532–6542. [doi: 10.1109/TII.2019.2945367]

[160] Brakerski Z, Gentry C, Vaikuntanathan V. (Leveled) fully homomorphic encryption without bootstrapping. ACM Trans. on Computation Theory, 2014, 6(3): 13. [doi: 10.1145/2633600]

[161] El Bansarkhani R, Dagdelen Ö, Buchmann J. Augmented learning with errors: The untapped potential of the error term. In: Proc. of the 19th Int'l Conf. on Financial Cryptography and Data Security. San Juan: Springer, 2015. 333–352. [doi: 10.1007/978-3-662-47854-7_20]

[162] Fang C, Guo YB, Hu YJ, Ma BW, Feng L, Yin AQ. Privacy-preserving and communication-efficient federated learning in Internet of Ihings. Computers & Security, 2021, 103: 102199. [doi: 10.1016/j.cose.2021.102199]

[163] Elgamal T. A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans. on Information Theory, 1985, 31(4): 469–472. [doi: 10.1109/TIT.1985.1057074]

[164] Froelicher D, Troncoso-Pastoriza JR, Pyrgelis A, Sav S, Sousa JS, Bossuat JP, Hubaux JP. Scalable privacy-preserving distributed learning. Proc. on Privacy Enhancing Technologies, 2021, 2021(2): 323–347. [doi: 10.2478/popets-2021-0030]

[165] Mouchet C, Troncoso-Pastoriza J, Bossuat JP, Hubaux JP. Multiparty homomorphic encryption from ring-learning-with-errors. Proc. on Privacy Enhancing Technologies, 2021, 2021(4): 291–311. [doi: 10.2478/popets-2021-0071]

[166] Sav S, Pyrgelis A, Troncoso-Pastoriza JR, Froelicher D, Bossuat JP, Sousa JS, Hubaux JP. POSEIDON: Privacy-preserving federated neural network learning. In: Proc. of the 28th Annual Network and Distributed System Security Symp. NDSS, 2021.

[167] Konečný J, McMahan B, Ramage D. Federated optimization: Distributed optimization beyond the datacenter. arXiv:1511.03575, 2015.

[168] Bonawitz K, Ivanov V, Kreuter B, Marcedone A, McMahan HB, Patel S, Ramage D, Segal A, Seth K. Practical secure aggregation for privacy-preserving machine learning. In: Proc. of the 2017 ACM SIGSAC Conf. on Computer and Communications Security. Dallas: ACM, 2017. 1175–1191. [doi: 10.1145/3133956.3133982]

[169] Niu CY, Wu F, Tang SJ, Hua LF, Jia RF, Lv CF, Wu ZH, Chen GH. Billion-scale federated learning on mobile clients: A submodel design with tunable privacy. In: Proc. of the 26th Annual Int'l Conf. on Mobile Computing and Networking. London: Association for Computing Machinery, 2020. 31. [doi: 10.1145/3372224.3419188]

[170] Li WQ, Milletarì F, Xu DG, Rieke N, Hancox J, Zhu WT, Baust M, Cheng Y, Ourselin S, Cardoso MJ, Feng A. Privacy-preserving federated brain tumour segmentation. In: Proc. of the 10th Int'l Workshop on Machine Learning in Medical Imaging. Shenzhen: Springer, 2019. 133–141. [doi: 10.1007/978-3-030-32692-0_16]

[171] Zhao B, Fan K, Yang K, Wang ZL, Li H, Yang YT. Anonymous and privacy-preserving federated learning with industrial big data. IEEE Trans. on Industrial Informatics, 2021, 17(9): 6314–6323. [doi: 10.1109/TII.2021.3052183]

[172] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: Proc. of the 32nd

Int'l Conf. on Machine Learning. Lille: ACM, 2015. 448–456. [doi: 10.5555/3045118.3045167]

[173] Andreux M, Du Terrail JO, Beguier C, Tramel EW. Siloed federated learning for multi-centric histopathology datasets. In: Proc. of the 2nd MICCAI Workshop on Domain Adaptation and Representation Transfer, and Distributed and Collaborative Learning. Lima: Springer, 2020. 129–139. [doi: 10.1007/978-3-030-60548-3_13]

[174] So J, Güler B, Avestimehr AS. Byzantine-resilient secure federated learning. IEEE Journal on Selected Areas in Communications, 2021, 39(7): 2168–2181. [doi: 10.1109/JSAC.2020.3041404]

[175] Chamikara MAP, Bertok P, Khalil I, Liu D, Camtepe S. Privacy preserving distributed machine learning with federated learning. Computer Communications, 2021, 171: 112–125. [doi: 10.1016/j.comcom.2021.02.014]

[176] Oh SJ, Schiele B, Fritz M. Towards reverse-engineering black-box neural networks. In: Samek W, Montavon G, Vedaldi A, Hansen LK, Müller KR, eds. Explainable AI: Interpreting, Explaining and Visualizing Deep Learning. Cham: Springer, 2019. 121–144. [doi: 10.1007/978-3-030-28954-6_7]

[177] Wang BH, Gong NZ. Stealing hyperparameters in machine learning. In: Proc. of the 2018 IEEE Symp. on Security and Privacy. San Francisco: IEEE, 2018. 36–52. [doi: 10.1109/SP.2018.00038]

[178] Bourtoule L, Chandrasekaran V, Choquette-Choo CA, Jia HR, Travers A, Zhang BW, Lie D, Papernot N. Machine unlearning. In: Proc. of the 2021 IEEE Symp. on Security and Privacy. San Francisco: IEEE, 2021. 141–159. [doi: 10.1109/SP40001.2021.00019]

[179] Xu GW, Li HW, Liu S, Yang K, Lin XD. VerifyNet: Secure and verifiable federated learning. IEEE Trans. on Information Forensics and Security, 2020, 15: 911–926. [doi: 10.1109/TIFS.2019.2929409]

附中文参考文献:

[1] 刘俊旭, 孟小峰. 机器学习的隐私保护研究综述. 计算机研究与发展, 2020, 57(2): 346–362. [doi: 10.7544/issn1000-1239.2020.20190455]

[12] 黄倩怡, 李志洋, 谢文涛, 张黔. 智能家居中的边缘计算. 计算机研究与发展, 2020, 57(9): 1800–1809. [doi: 10.7544/issn1000-1239.2020.20200253]

[20] 张思思, 左信, 刘建伟. 深度学习中的对抗样本问题. 计算机学报, 2019, 42(8): 1886–1904. [doi: 10.11897/SP.J.1016.2019.01886]

[94] 王鹃, 樊成阳, 程越强, 赵波, 韦韬, 严飞, 张焕国, 马婧. SGX技术的分析和研究. 软件学报, 2018, 29(9): 2778–2798. http://www.jos.org.cn/1000-9825/5594.htm [doi: 10.13328/j.cnki.jos.005594]

[102] 李欣姣, 吴国伟, 姚琳, 张伟哲, 张宾. 机器学习安全攻击与防御机制研究进展和未来挑战. 软件学报, 2021, 32(2): 406–423. http://www.jos.org.cn/1000-9825/6147.htm [doi: 10.13328/j.cnki.jos.006147]

[107] 纪守领, 杜天宇, 李进锋, 沈超, 李博. 机器学习模型安全与隐私研究综述. 软件学报, 2021, 32(1): 41–67. http://www.jos.org.cn/1000-9825/6131.htm [doi: 10.13328/j.cnki.jos.006131]

[108] 谭作文, 张连福. 机器学习隐私保护研究综述. 软件学报, 2020, 31(7): 2127–2156. http://www.jos.org.cn/1000-9825/6052.htm [doi: 10.13328/j.cnki.jos.006052]

[124] 张钰, 刘建伟, 左信. 多任务学习. 计算机学报, 2020, 43(7): 1340–1378. [doi: 10.11897/SP.J.1016.2020.01340]

[140] 董业, 侯炜, 陈小军, 曾帅. 基于秘密分享和梯度选择的高效安全联邦学习. 计算机研究与发展, 2020, 57(10): 2241–2250. [doi: 10.7544/issn1000-1239.2020.20200463]

[142] 熊平, 朱天清, 王晓峰. 差分隐私保护及其应用. 计算机学报, 2014, 37(1): 101–122. [doi: 10.3724/SP.J.1016.2014.00101]

[151] 李宗育, 桂小林, 顾迎捷, 李雪松, 戴慧珺, 张学军. 同态加密技术及其在云计算隐私保护中的应用. 软件学报, 2018, 29(7): 1827–1851. http://www.jos.org.cn/1000-9825/5354.htm [doi: 10.13328/j.cnki.jos.005354]

顾育豪(1993－), 男, 博士生, 主要研究领域为深度学习安全和隐私.

白跃彬(1962－), 男, 博士, 教授, 博士生导师, CCF 高级会员, 主要研究领域为新型智能计算系统, 云操作系统, 嵌入式操作系统.