软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software, 2022,33(3):1057−1092 [doi: 10.13328/j.cnki.jos.006446]

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

联邦学习中的隐私保护技术*

刘艺璇[1,2], 陈 红[1,2], 刘宇涵[1,2], 李翠平[1,2]

[1](中国人民大学 信息学院, 北京 100872)

[2](数据工程与知识工程教育部重点实验室(中国人民大学), 北京 100872)

通信作者: 陈红, E-mail: chong@ruc.edu.cn

摘 要: 联邦学习是顺应大数据时代和人工智能技术发展而兴起的一种协调多个参与方共同训练模型的机制. 它允许各个参与方将数据保留在本地, 在打破数据孤岛的同时保证参与方对数据的控制权. 然而联邦学习引入了大量参数交换过程, 不仅和集中式训练一样受到模型使用者的威胁, 还可能受到来自不可信的参与设备的攻击, 因此亟需更强的隐私手段保护各方持有的数据. 分析并展望了联邦学习中的隐私保护技术的研究进展和趋势. 简要介绍联邦学习的架构和类型, 分析联邦学习过程中面临的隐私风险, 总结重建、推断两种攻击策略, 然后依据联邦学习中的隐私保护机制归纳隐私保护技术, 并深入调研应用上述技术的隐私保护算法, 从中心、本地、中心与本地结合这 3 个层面总结现有的保护策略. 最后讨论联邦学习隐私保护面临的挑战并展望未来的发展方向.

关键词: 联邦学习; 隐私保护; 隐私攻击; 差分隐私; 同态加密; 安全计算

中图法分类号: TP181

中文引用格式: 刘艺璇, 陈红, 刘宇涵, 李翠平. 联邦学习中的隐私保护技术. 软件学报, 2022, 33(3): 1057–1092. http://www.jos.org.cn/1000-9825/6446.htm

英文引用格式: Liu YX, Chen H, Liu YH, Li CP. Privacy-preserving Techniques in Federated Learning. Ruan Jian Xue Bao/Journal of Software, 2022, 33(3): 1057−1092 (in Chinese). http://www.jos.org.cn/1000-9825/6446.htm

Privacy-preserving Techniques in Federated Learning

LIU Yi-Xuan[1,2], CHEN Hong[1,2], LIU Yu-Han[1,2], LI Cui-Ping[1,2]

[1](School of Information, Renmin University of China, Beijing 100872, China)

[2](Key Laboratory of Data Engineering and Knowledge Engineering (Renmin University of China), Ministry of Education, Beijing 100872, China)

Abstract: With the era of big data and the development of artificial intelligence, Federated learning (FL) emerges as a distributed machine learning approach. It allows multiple participants to train a global model collaboratively while keeping each of their training datasets in local devices. FL is created to break up data silos and preserve the privacy and security of data. However, there are still a large number of privacy risks during data exchange steps, where local data is threatened not only by model users as in centralized training but also by any dishonest participants. It is necessary to study technologies to achieve rigorous privacy-preserving approaches. The research progress and trend of privacy-preserving techniques for FL are surveyed in this paper. At first, the architecture and type of FL are introduced, then privacy risks and attacks are illustrated, including reconstruction and inference strategies. According to the mechanism of privacy preservation, the main privacy protection technologies are introduced. By applying these technologies, privacy defense strategies are presented and they are abstracted as 3 levels: local, central, local & central. Challenges and future directions of privacy-preserving in federated learning are discussed at last.

Key words: federated learning; privacy-preserving; privacy attack; differential privacy; homomorphic encryption; secure computation

* 基金项目: 国家重点研发计划(2018YFB1004401); 国家自然科学基金(62072460, 62076245, 61772537, 61772536, 62172424); 北京市自然科学基金(4212022); 中国人民大学科学研究基金(中央高校基本科研业务费专项资金资助) (21XNH180)

本文由“数据库系统新型技术”专题特约编辑李国良教授、于戈教授、杨俊教授和范举教授推荐.

收稿时间: 2021-06-30; 修改时间: 2021-07-31; 采用时间: 2021-09-13; jos 在线出版时间: 2021-10-21

近年来, 大数据驱动的人工智能迸发出巨大潜力, 在金融、医疗、城市规划、自动驾驶等多个领域完成了大规模复杂任务学习. 机器学习作为人工智能的核心技术, 其性能和隐私性也广受关注. 传统的机器学习需要由服务商收集用户的数据后集中训练, 但是用户的数据与用户个体紧密相关, 可能直接包含敏感信息, 如个人年龄、种族、患病信息等; 也可能间接携带隐含的敏感信息, 如个人网页浏览记录、内容偏好所隐含的用户政治倾向. 如果这些敏感信息在收集过程中被服务商泄露或者利用, 将直接威胁用户的人身安全、个人名誉和财产安全. 即便服务商没有直接公开用户数据, 集中训练后发布的模型也可能因为受到隐私攻击而泄露参与训练的数据. 随着隐私问题受到的关注程度日益提高, 用户分享数据的意愿越来越低. 与之矛盾的是, 人工智能技术却必须依靠大量数据收集和融合, 如果不能获取完整丰富的信息来训练模型并发展技术, 人工智能应用的发展将受到严重限制.

在数据孤岛现象与数据融合需求的矛盾逐渐凸显的背景下, 联邦学习(federated learning, FL)应运而生. 2017 年, Google 公司首次提出了联邦学习的概念[1], 这是一种由多个数据持有方(如手机、物联网设备, 或者金融、医疗机构等)协同训练模型而不分享数据, 仅在中间阶段交换训练参数的学习机制. 理想状况下, 联邦学习得到的共享模型与数据集中在中心服务器上训练所得模型相比, 效果相近或更好[2]. 由此, 企业能够通过合法且高效的方式融合数据提取信息, 个人或其他持有数据的机构依然能够在享受企业提供的人工智能服务的同时, 保有数据的控制权.

尽管联邦学习避免了将数据直接暴露给第三方, 对于数据隐私有天然的保护作用, 但是其中依然存在大量隐私泄露的风险.

- 首先, 联邦学习需要交换中间参数协同训练, 可能泄露隐私. 与集中式学习不同, 联邦学习训练过程需要交换大量中间参数, 其所携带原始数据会暴露在所有参与训练的角色面前, 带来泄露的风险. 例如, 已有研究表明, 可以通过梯度还原部分原始数据[3], 或根据中间参数推断掌握的记录内容是否来自某个特定参与者[4].
- 其次, 不可靠的参与方加剧了隐私泄露的风险. 联邦学习中, 各个参与方由于地理、设备等条件不同, 通信内容的有效性和身份的真实性都难以确认, 因此一旦出现不可靠的参与方攻击, 极易泄露隐私. 例如, 半诚实的参与方能够根据合法获取的中间参数推断出其他参与方的标签或数据; 而恶意的参与方更进一步, 能够通过上传精心设计的有害信息诱导其他参与方暴露更多自身数据, 或者不遵守隐私协议进而影响全局的隐私性.
- 此外, 训练完成的模型也面临着隐私泄露的风险. 即便联邦学习的过程中参数没有泄露, 直接发布训练所得的模型依然存在极大风险. 这种风险来自机器学习自身的脆弱性. 在训练中, 模型提高准确性依赖于对数据样本的规律挖掘. 但是研究者[4]注意到, 追求模型在训练样本上的准确度, 可能导致模型的参数乃至结构“记住”训练样本的细节, 使得模型携带训练集的敏感信息. 根据这一特性, 攻击者可以通过反复查询模型的预测接口来推测某条记录是否存在于训练集、推测模型的具体参数, 而根据模型发布的参数能够进一步推测训练集成员或训练集具体样本.

由此可见, 不加保护的进行联邦学习, 训练中涉及的众多参与者的数据都将面临泄露的风险. 而数据一旦泄露, 不仅隐私泄露者面临严重损失, 参与者间彼此信任合作的联合训练模式也将难以为继.

解决联邦学习信息泄露问题迫在眉睫. 然而, 联邦学习中数据分布复杂、应用场景丰富且需要多次数据交换, 这些因素为隐私保护带来一系列挑战.

- 第一, 联邦学习的训练场景多样且需求复杂, 现有的隐私保护方法无法通用. 已有的集中式机器学习隐私保护研究以中心服务器诚实为前提, 仅考虑模型发布后可能受到的攻击, 没有针对内部攻击者的解决方案. 而且现有算法大多针对单一的集中式训练场景, 没有考虑多个参与方、多种架构、多种数据分布方式下的数据交换和模型协同训练的情况. 因此, 设计适应不同场景和不同需求的隐私保护算法, 同时抵御外部和内部攻击, 是联邦学习隐私保护的重要挑战.
- 第二, 联邦学习中参与方的可信程度低, 潜在的攻击角度多, 对隐私保护算法的鲁棒性要求更高. 这

里, 鲁棒性指模型容忍恶意攻击稳定运行的能力. 联邦学习中, 参与者一旦发起攻击, 能够观察到更多的中间参数, 甚至能够篡改参数影响训练过程, 隐私防御的难度远高于外部出现的攻击. 而参与者之间如果共谋, 可能获取更多敏感信息. 因此, 提高隐私保护算法的鲁棒性, 减少隐私算法中对参与者的可信程度的假设, 是联邦学习隐私保护面临的难题.

- 第三, 联邦学习本身通信不稳定, 模型计算代价高, 因而对隐私保护机制的通信量和复杂度要求严格. 现实场景下的联邦学习所面临的复杂松散的网络结构导致终端通信不稳定, 在此基础上的隐私保护算法难以简化. 而复杂的隐私保护算法将带来更高的计算量、更大通信代价, 进一步制约联邦学习的训练效率. 研究高效率、轻量级的联邦学习隐私保护算法, 降低额外开销, 是联邦学习隐私保护必须面对的挑战.
- 第四, 联邦学习中参数维度高、数据分布不均, 难以在提供隐私保护的同时保持模型的可用性. 联邦学习中间参数的维度与模型结构和输入数据维度相关, 参数维度往往极高, 造成了极大的隐私开销. 此外, 联邦学习的用户数量不定且数据集大小不一, 如何在平衡不同数据集的同时保护隐私, 也是一个巨大挑战.

综上所述, 更加精细的隐私策略设计、更加精确的隐私预算分配、更加适应数据交换的隐私协议构建, 是联邦学习隐私保护进一步发展必须面对的议题. 而明确现有的隐私问题和保护手段, 是技术发展的基础. 联邦学习的基础——机器学习的隐私攻击和防御已经被充分调研[5]. 机器学习面临的外部攻击同样威胁着联邦学习的发布模型, 但是机器学习的隐私保护手段却远远不足以为联邦学习提供保护. 这是由于联邦学习同时面临着传统的外部攻击和其独有的内部攻击, 因此联邦学习的隐私保护方案必须同时为内部训练过程和外部模型发布提供双重保护.

另外, 已有学者调研了联邦学习隐私保护的现状, 但由于思路与本文不同, 侧重的方法和文献也不相同. Lyv 等人[6]和 Wang 等人[7]对联邦学习可能受到的攻击作了详细的阐述, 但是在安全攻击和隐私攻击的区分上没有进一步调研. 本文明确两种攻击的概念范围: 以窃取数据、破坏模型隐私性和机密性为目的的攻击为隐私攻击, 以干扰模型训练结果、破坏模型可用性和完整性的攻击为安全攻击. 此外, 本文还依据现有的隐私攻击技术的原理归纳了主要策略分类. 现有文献[7−10]均从技术或训练阶段的角度分析了目前的联邦学习隐私保护算法, 而本文根据联邦学习自身特性分析其特有的隐私泄露内容和泄露位置, 从隐私保护的对象的角度出发建立分类框架, 并归纳每个类别中主要的保护机制, 进而分析采用不同技术的算法的共性并探究机制本身的优势和不足. 进一步地, 本文建立了攻击策略与保护机制之间的联系, 并在此基础上尝试为联邦学习隐私保护的发展提出建议.

本文第 1 节介绍联邦学习的架构和类型, 以及相应场景下的训练方式. 第 2 节分析联邦学习面对的隐私泄露风险来源, 总结具体的攻击策略. 第 3 节介绍多种隐私保护技术原理, 并将其归纳为信息模糊、过程加密两种隐私保护机制. 第 4 节调研隐私保护技术在联邦学习中的应用, 涵盖本地保护、中心保护、中心与本地结合这 3 种保护策略, 并对每种策略展开更加详细的阐述. 第 5 节讨论现有不足并展望未来方向.

1 联邦学习

联邦学习的一般定义为[11]: N 个参与方$\{F_1,\ldots,F_N\}$各自持有训练集$\{D_1,\ldots,D_N\}$. 联邦学习中, 各个参与方在不将本地数据 D_i 暴露给第三方的情况下, 协作训练模型 M_{FED}. 为了给联邦学习模型一个衡量标准, 设传统的集中式机器学习将各个数据集收集合并为 $D=D_1\cup\ldots\cup D_N$ 以训练模型 M_{SUM}. 令 V_{FED} 为联邦学习模型 M_{FED} 精度(performance), V_{SUM} 为传统机器学习模型 M_{SUM} 精度. 存在非负实数δ, 使得:

$$|V_{FED}-V_{SUM}|<\delta,$$

则称此联邦学习模型具有δ的精度损失. 可见, 使各个数据集留在本地协同训练所得模型的精度, 理想状况下应当接近于将数据集集中后训练所得模型的精度.

区别于传统的分布式机器学习, 联邦学习具有如下特点.

(1) 各个参与方的训练集非独立同分布. 各个参与方仅掌握局部信息, 其分布与全局不一定相同; 各个参与方仅掌握整个数据集的部分属性及标签信息, 且各方之间属性和标签可能不完全重叠.

(2) 各个参与方的训练集大小可能不平衡. 某些参与方可能由于其规模、影响力等因素掌握更多数据.

(3) 参与方数量不定. 参与者数量可能很少, 例如只有几个企业交换数据集; 也可能极多, 如训练涉及数以万计的 App 使用者.

(4) 通信受限. 与分布式相比, 联邦学习的架构更为松散, 参与的设备可能存在频繁掉线、通信缓慢等情况, 因此联邦学习的通信代价同样受到极大关注.

根据这些特点, 学者为联邦学习设计了不同的架构方式和学习类型.

1.1 联邦学习架构

常见的联邦学习架构为客户-服务器. 典型的客户-服务器架构由一个中心服务器和多个持有数据的客户端组成. 被广泛采用的联邦平均 FedAvg[1]即是基于客户-服务器架构设计的算法. 在训练中, 中心服务器将随机初始化的模型结构和参数分发给客户端, 客户端根据本地数据训练并更新模型后将参数上传. 中心服务器收到各方参数后聚合计算, 更新模型参数再次下发. 该过程循环, 直至模型收敛或训练终止. 除了常见的模型参数交换以外, 也存在梯度交换、数据特征的嵌入式表示交换等方式. 在此架构下, 原始数据不需要传输, 但是本地中间参数暴露给了中心服务器, 全局中间参数则会被每个客户端获取, 数据交换过程中, 巨大的通信量也会影响训练效率. 而当参与训练的客户端数量过多时, 中心服务器的聚合计算甚至可能成为全局训练效率的瓶颈.

当没有中心服务器时, 联邦学习采用另一种常见架构: 端对端的网络架构[12]. 这种架构仅由持有数据的终端组成. 参与训练的终端 F_i 直接将训练参数发送给下一个(或多个)终端 F_{i+1}, 下一个(或多个)终端 F_{i+1} 在收集到的一个(或多个)参数基础上继续训练, 直到模型收敛或者训练终止. 端对端网络架构不依赖中心服务器这样的第三方机构, 本地中间参数直接在参与方之间传送. 因此需要考虑参与方如何协商使用相同的模型、算法、初始化参数等基本信息, 协调各方参与训练的顺序.

为了下文中概念统一、表述清晰, 本文将客户-服务器中的服务器称为中心服务器; 将客户-服务器中的客户端和端对端架构中的参与训练终端统称为终端; 所有参与训练的服务器、终端统称为参与方. 训练过程中发送的梯度、模型参数、嵌入式表示等, 统称为中间参数. 上述两种典型架构如图 1 所示.

图 1 联邦学习中的典型架构

1.2 联邦学习类型

根据参与方的样本分布情况, 联邦学习按照数据的划分情况可以分为 3 种类型: 横向联邦学习、纵向联邦学习、迁移联邦学习. 不同的数据的划分方式需要的训练方式和中间参数不同, 也为隐私泄露的风险和保护方式带来影响.

横向联邦学习中, 各个参与方持有的数据特征相同, 但掌握的样本不同. 例如, 几个不同城市的医院可能掌握着不同病人的情况, 但是由于具备相似的医疗手段, 医院获取属性的属性相同. 横向联邦学习中典型的方式之一是第 1.1 节所描述的联邦平均算法 FedAvg, 包括梯度平均和模型平均两种类型[13], 多由客户-服务器架构实现. 梯度平均是指终端交换和聚合模型梯度, 而模型平均指聚合模型参数. 在端对端架构中, 各个参与方训练本地模型, 通过循环发送给下一个(或多个)训练方或者随机传输某个(或多个)终端[14]实现模型参数的共享.

而纵向联邦学习则针对相反的情形, 即各个参与方持有的数据特征不同, 但掌握的样本相同. 例如, 同一个城市中的医院和银行都接待过同一个市民, 保留着该市民的就诊记录或资金状况. 显然, 医院和银行获取的数据属性完全不同, 但是所持有的样本 ID 是重叠的. 纵向联邦学习首先需要参与方对齐相同 ID 的样本, 然后, 各个参与方在对齐的样本上分别训练本地模型并分享参数. 不同架构同样都适用于纵向联邦学习, 但由于数据的纵向分布, 参与方之间的依赖程度更高, 模型需要更加精细地设计. 纵向联邦学习已应用于线性回归[11]、提升树[15]、梯度下降[16]等多种模型上. 以纵向联邦学习线性回归算法[11]为例, 该算法在样本对齐后, 将损失函数的梯度拆分, 使得两个参与方能够使用各自的本地数据分别计算梯度的一部分, 而需要共同计算的部分则通过双方交换参数协同完成. 纵向分布的数据之间紧密的相关性, 为纵向学习的效率和容错性带来挑战.

上述两种类型都是比较理想的情况, 现实生活中, 大部分参与方所持有的数据, 在特征和样本 ID 上的重叠都比较少且数据集分布不平衡. 针对这样的情形, 迁移学习被应用到联邦学习中来. 迁移学习作为一种有效的学习思想, 能够将相关领域中的知识迁移到目标领域中, 使得各个参与方共同学习得到迁移知识. 以两方迁移学习为例[17], 假设一方 A 掌握样本的部分特征和全部标签, 另一方 B 掌握部分特征, 双方特征和样本 ID 之间都有少量重叠. 联邦迁移学习首先对齐样本并共同训练模型, 然后预测 B 方样本的标签. 为了达到预期效果, 训练的目标函数包含两个部分: 一部分是根据已有的标签预测 B 方样本, 使预测误差最小化; 另一部分是 A 与 B 对齐的样本之间的嵌入式表示的区别最小化. 各方根据目标函数在本地训练, 并交换中间参数更新模型, 直至模型收敛.

目前, 纵向和迁移联邦学习的隐私保护算法研究还不成熟, 且保护方式与横向联邦学习场景类似. 为了表述简洁, 下文中调研的隐私保护算法若无特别说明, 即为横向联邦学习场景.

2 联邦学习中的隐私泄露风险

尽管联邦学习不直接交换数据, 比传统的集中式机器学习训练有了更高的隐私保障, 但联邦学习本身并没有提供全面充分的隐私保护, 依然面临着信息泄露的威胁. 模型面临的隐私泄露风险来自模型训练自身的脆弱性和攻击者的强大能力: 模型训练过程中, 独特架构和训练阶段决定了隐私泄露的位置和时机; 攻击者的角色和能力, 决定了隐私泄露的内容和程度. 而攻击者依据自身特性所采取的攻击策略, 则进一步影响攻击者的能力, 从而影响模型隐私泄露的风险. 理清隐私泄露的风险, 才能为联邦学习隐私防御找到总体方向.

2.1 隐私泄露风险来源

为了在下文中更好地描述隐私攻击, 我们首先建立联邦学习攻击模型.

- 根据角色, 攻击者分为内部和外部: 内部攻击者包括掌握训练的中间参数并且参与训练过程的终端和中心服务器; 而外部攻击者包括掌握模型发布的参数及查询接口但没有参与训练过程的模型使用者. 与外部攻击者相比, 内部攻击者掌握模型的更多信息, 攻击能力更强.
- 根据可信程度, 攻击者分为半诚实角色和恶意角色: 半诚实角色指参与方严格遵守训练协议和流程, 仅根据合法获取的信息分析推断, 对于训练结果没有影响; 恶意角色指参与方不遵守协议, 在参与过程中恶意篡改数据、注入模块, 诱导目标泄露隐私并影响训练结果.
- 根据攻击模式, 攻击分为被动和主动: 被动攻击指攻击者仅观察或访问模型获取信息; 主动攻击指攻击者篡改数据或模型, 参与并影响训练过程. 需要说明的是, 攻击模式与可信程度并非完全对应. 存

在少数主动攻击者能够在修改上传参数诱导目标泄露隐私的同时不影响联邦训练目标, 诚实正确地完成训练任务.

- 根据攻击者知识, 攻击分为白盒攻击和黑盒攻击: 白盒攻击指攻击者掌握模型的相关信息, 包括数据的分布和统计信息、模型训练完成的结构参数或模型训练过程中的中间参数; 黑盒攻击指攻击者对相关信息一无所知, 仅有请求查询的权限.

联邦学习包含参数上传、下发、参数传输、模型发布等多个阶段, 其中, 参数上传、下发为客户-服务器架构所特有的阶段, 参数传输为端对端架构所特有的阶段, 模型发布为两种架构都有的阶段. 每个阶段隐私泄露的位置和内容不同, 威胁隐私的攻击者角色也不相同. 研究联邦学习不同阶段隐私泄露风险, 有助于为隐私保护提供清晰的思路和方向. 隐私泄露风险来源的对比见表 1.

表 1 隐私泄露风险来源

联邦学习自身脆弱性				潜在攻击者	
训练架构	训练阶段	泄露位置	泄露内容	属性	内外
客户-服务器	参数下发	中心	全局参数	终端	内部
	模型发布		训练完成的模型	外部使用者	外部
端对端	参数上传	本地	本地参数	中心服务器	内部
	参数传输			其他终端	
	模型发布		训练完成的模型	外部使用者	外部

在客户-服务器架构下, 训练分为 3 个阶段.

- 第 1 阶段, 本地(终端)训练后, 上传本地参数给中心服务器. 此阶段的潜在攻击者多为中心服务器. 中心服务器根据收集的本地参数能够发起重建攻击, 恢复目标终端的原始数据; 或者发起属性推断攻击, 推断目标终端的数据中是否含有某些敏感属性.
- 第 2 阶段, 中心服务器聚合各个终端的参数处理后, 再次下发全局参数. 此时的潜在攻击者是不可信的终端. 终端能够根据全局中间参数发起攻击重建某个类别样本(generic sample), 或推断某条记录的敏感属性是否存在, 进而根据训练集的共同特征推断拥有该记录的个体的情况, 例如训练集是艾滋病患者基因数据, 如果该个体属于该训练集, 则一定也患有艾滋病.
- 第 3 阶段, 模型训练完成, 由中心服务器发布模型. 一般的发布方式包括直接将模型部署在用户端, 或提供 API 访问接口两种. 此时的潜在攻击者是外部使用者. 不可信的外部使用者能够根据模型的参数或模型的预测结果发起推断攻击、重建攻击或参数提取攻击[18−22]. 通过发布的模型推测参与训练的数据集样本, 或根据 API 访问接口推测模型参数, 进而攻击训练数据.

在端对端架构下, 训练分为两个阶段:

- 第 1 阶段, 本地(终端)训练后, 将本地参数传输给下一终端. 攻击者为不可信的接收终端. 终端接收本地参数后, 同样能够发起重建攻击或属性推断攻击, 致使本地的原始数据泄露.
- 第 2 阶段, 模型训练完成后, 终端若发布模型, 则面临与客户-服务器架构同样的外部攻击; 若仅供内部使用, 则无须考虑.

需要说明的是, 联邦学习的隐私攻击主要由内部参与方发起. 与外部攻击者相比, 内部攻击者具备更强的能力, 不仅可以在训练过程中通过直接获取数据交换中的特征嵌入式表示、梯度和模型参数等发起攻击, 还能够通过替换样本、更改梯度甚至修改损失函数等方式影响模型的训练过程, 诱导目标终端暴露更多隐私信息, 完成推断攻击和重建攻击. 联邦学习为了协同训练和共享模型需要更多参与者, 却缺乏与之对应的身份确认机制和诚信保障, 难以防范“内部”泄露. 传统的集中式机器学习隐私保护能够抵御外部攻击, 却没有抵御内部攻击的能力. 为了理清联邦学习面对的风险, 本文首先介绍外部攻击作为基础, 重点针对其特有的内部隐私攻击展开分析.

2.2 隐私攻击策略

根据上述的隐私泄露风险和攻击者的能力, 研究者设计了不同的联邦学习架构及阶段下可能的隐私攻

击, 并通过实验展现了这些攻击对敏感数据的巨大威胁. 隐私攻击者包括参与模型训练的内部角色和未参与模型训练仅能接触发布模型的外部角色. 攻击者采取的主要策略有重建攻击和推断攻击: 重建攻击中, 攻击者根据掌握的中间参数以及模型信息恢复部分训练数据; 推断攻击中, 攻击者根据中间参数和发布参数推断训练集中是否含有特定的记录. 内部以及外部攻击者采用上述策略对联邦学习发起隐私攻击, 获取训练集的敏感信息. 其中, 内部隐私攻击方案的总结参见表 2.

表 2 内部隐私攻击具体方案

文献	攻击目标			攻击者能力		攻击策略		优缺点		攻击者知识
	攻击对象	架构	数据分割	角色	可信度	模式	策略	优点	缺点	背景知识
[23]	全局梯度	客户-服务器	横向	终端	半诚实	被动	类别重建	能够攻击类别、身份、真实与生成数据等多种信息; 对攻击者知识要求低	无法重建单个样本; 计算量较大	其他标签
[24]	局部梯度			服务器	恶意	主动				测试集
[25]					半诚实	被动				
[3]		端对端		终端	半诚实	被动	样本重建	对攻击者能力要求低; 对攻击者知识要求低; 能重建单个样本	无法攻击泛化良好的模型; 无法攻击由多个样本生成的梯度	无
[26]		客户-服务器		服务器						
		端对端		终端						
[27]		客户-服务器		服务器						
[28]		客户-服务器/端对端	横向/纵向	服务器/终端						预训练模型
[29]										
[30]		客户-服务器	横向	服务器		主动	属性推断	能够推断与训练任务无关的属性	无法推断敏感属性所属样本; 需要辅助集	有监督辅助集
[31]	全局梯度			终端	半诚实	被动	成员推断	攻击准确性较高; 对泛化良好的模型有一定的攻击能力	无法获取完整样本信息; 辅助集生成过程复杂低效	无监督辅助集
	局部梯度				恶意	主动				
					半诚实	被动				

2.2.1 重建攻击

重建攻击(reconstruction attack)指攻击者根据训练中间参数、模型的参数或者请求查询所得输出, 恢复参与训练的数据集中的信息. 根据攻击者角色, 重建攻击分为外部和内部攻击: 外部重建攻击是在模型训练完成并发布后, 外部使用者发起的攻击; 内部重建攻击则是在模型训练阶段, 内部参与方发起的攻击.

外部重建攻击中, 攻击者仅能掌握模型的查询结果或模型发布的结构和参数, 因此只能不断试探模型的输出结果, 通过调整输入数据使输出值向预期方向靠拢. Fredrikson 等人[20]首次设计了外部攻击者在黑盒情况下发起的模型倒推攻击, 该算法基于模型的输出和一些非敏感属性恢复了病人的基因信息. 攻击者假设样本共有 d 维特征, 其中, f_1 到 f_{d-1} 为非敏感特征, 在给定非敏感特征和模型输出时, 最大化敏感特征 f_d 的后验概率. 上述工作仅能推断敏感属性, Fredrikson 等人[21]在随后的工作中设计了白盒情况下的模型倒推攻击. 外部攻击者根据训练完成的模型参数训练深度学习模型, 恢复训练集中的全部特征. 该攻击通过保持网络结构和参数不变, 对输入的随机像素值和随机标签梯度下降. 当模型预测置信度达到最优时, 生成的图片像素与训练集数据高度近似. Luo 等人[32]首次提出了纵向联邦学习中的特征重建攻击. 掌握标签的主动参与方在获取最终模型的预测结果后, 能够根据公式倒推或通过路径限制重建不掌握标签的被动参与方的数据特征. 上述算法的攻击者角色均为外部攻击者, 仅能掌握模型得查询结果或模型发布的结构和参数, 因此只能不断输入数据试探模型的输出结果.

外部攻击以模型输出以及发布的最终参数为依据, 重建整个数据集的泛化样本(generic sample), 难以获取详细的敏感信息. 此外, 借助有效的泛化、降低输出精度等手段, 即可在很大程度上抵御这类攻击. 相比之下, 联邦学习的内部重建攻击具有更加丰富的知识背景, 能够以中间参数为依据发起攻击. 中间参数不仅与

用户数据紧密相关, 而且在迭代中多次暴露. 内部攻击者能够据此重建特定用户的具体敏感信息. 在主动攻击的情况下, 还能够通过修改中间参数、上传有害信息来影响模型的训练过程, 甚至诱导隐私泄露. 具体的内部重建攻击包含两种类型: 类别重建和样本重建.

内部类别重建是重建攻击中的常见类型, 该攻击通过重建某个类别的通用样本模式获取目标类别(target class)中的敏感信息. 例如, 在训练图片识别分类器时, 训练集的一个类别中包含的图片主体是一致的, 则类别重建能够恢复目标类别中主体的共性信息. Hitaj 等人[23]针对客户-服务器架构的联邦学习, 提出了基于生成对抗网络(generative adversarial networks, GAN)的主动重建攻击. 攻击者作为参与训练的终端, 上传篡改参数给服务器诱导其他诚实终端暴露信息, 从而推测仅由目标终端掌握的类别的样本. 具体来说, 假设目标终端拥有类别[a,b]的样本, 攻击者拥有类别[b,c]的样本. 为获取目标类别 a 的信息, 攻击者首先在本地训练生成对抗网络, 利用从中心服务器获取的全局梯度更新其判别器后生成目标类别 a 的近似样本, 然后将近似样本故意标注为类别 c 训练本地分类器, 并上传参数到中心服务器. 在迭代过程中, 由于攻击者故意将真实标签为 a 的样本分类到 c 中, 目标终端需要暴露更多与 a 相关信息来“纠正”全局梯度, 这些信息使得攻击者的生成对抗网络获得更加准确的参数, 恢复的样本信息比 Fredrikson 等人[20]面向模型输出的攻击结果更加丰富、清晰. 但是这种算法要求攻击者主动篡改模型参数来影响全局模型的训练结果, 对于攻击者的能力假设过强. 而篡改后的参数影响力也会在聚合平均时被稀释, 导致攻击者的能力不能得到完全发挥. Wang 等人[24]随后提出了一种被动攻击模型 mGAN-AI, 攻击者能够在不干扰训练过程的情况下达到较好的攻击效果. 具体来说, mGAN-AI 假设半诚实的中心服务器能够根据每次迭代所获取的各个终端的局部参数更新训练多任务生成对抗网络. 它利用目标终端的参数更新样本判别器, 以区分目标终端数据的样本类别、生成目标终端的近似数据; 利用其他终端的参数训练身份判别器, 以区分目标终端和其他终端的身份; 利用辅助数据集添加噪声训练真实数据判别器, 以区分真实数据和生成数据. 该模型能够在重建类别的同时不干扰模型的正常训练过程, 攻击手段更加隐蔽. 但是上述攻击中的身份识别环节假设中心服务器知晓各局部参数所对应的终端身份, 如果终端匿名上传参数则攻击失效. 针对此问题, Song 等人[25]基于 Orekondy 等人[33]的链接攻击思想, 进一步提出了匿名环境下的 mGAN-AI 攻击模型. 半诚实的中心服务器根据终端上传的匿名参数生成对应的参数表示(parameter representative), 通过衡量本轮的所有参数表示与上一轮的所有参数表示的相似度进行匹配, 相似的参数表示即属于同一个终端. 为使参数表示更加全面地表征此次更新, 参数表示的计算由终端梯度和终端模型参数同时参与完成. 为了准确衡量参数表示的相似度的同时生成合适的终端身份表示(identification representative), 攻击者辅助集训练卷积孪生网络融合身份识别和相似匹配两个训练目标, 以学习更有区分度的终端身份表示.该模型能够在匿名更新的环境下识别终端上传的参数, 并重建终端的泛化样本.

但是, 类别重建仍然存在一些局限: 首先, 它不能还原目标类别中的不同样本, 只适用于一个类别中的样本都类似的场景, 所能获取的敏感信息有限; 其次, 基于生成对抗网络的重建方式对攻击者的计算能力要求较高, 在手机终端等场景下并不适用.

内部样本重建相比于内部类别重建是一种更加精确的攻击方式, 能够恢复出一个类别中的多个样本, 提取每个样本的敏感信息. Zhu 等人[3]提出了深度泄露算法(DLG), 攻击者只需要获取目标终端的某一轮梯度, 通过有限的迭代来优化随机输入值, 使得生成梯度近似于目标梯度, 优化后的输入值即为生成的近似样本. 只要掌握局部梯度, 客户-服务器架构中的中心服务器和端对端架构中的参与方均可发起 DLG 攻击. 对于单个目标样本, 攻击是非常直观的. 攻击者首先随机生成一个样本的特征 x_1' 和标签 y_1', 计算样本的梯度 $\nabla W_i'$, 然后以该梯度 $\nabla W_i'$ 与目标终端的梯度∇W 之间的欧式距离 D 最小为优化目标, 更新样本:

$$x_{i+1}' \leftarrow x_i' - \eta \nabla_{x_i} D_i,\ y_{i+1}' \leftarrow y_i' - \eta \nabla_{y_i} D_i \tag{1}$$

其中, i 表示攻击的迭代轮数; η表示攻击的学习率. 而当重建目标为一个批次(batch)的样本时, 需考虑不同的样本之间可能混淆. 攻击者在更新样本时, 必须假设批次中样本有序, 然后处理每个样本. DLG 算法在更新样本时采用了梯度下降的方式, 因此能够在较少的迭代次数下恢复出原始样本. 然而, 由于初始的标签生成是完全随机的, DLG 算法在查找到标签真实值上的收敛速度还是有一定的随机性. Zhao 等人[26]基于 DLG 优化初

始标签的生成方式, 提出了 iDLG 算法. iDLG 首先针对梯度 ∇W_L^i 作出如下变形:

$$\nabla W_L^i = \frac{\partial l(x,c)}{\partial W_L^i} = \frac{\partial l(x,c)}{\partial y_i} \cdot \frac{\partial y_i}{\partial W_L^i} = g_i \cdot a_{L-1} \tag{2}$$

其中, $l(x,c)$表示损失函数, c 表示真实标签, g_i 为损失函数关于前向传播输出 y_i 的导数, a_{L-1} 表示神经网络第 $L-1$ 层的输出. 由于激活函数的存在, $a_{L-1}>0$, 因此, 梯度的正负仅与 g_i 相关. 而 g_i 仅在真实标签上的值为负数:

$$g_i = \frac{\partial l(x,c)}{\partial y_i} = \begin{cases} -1 + \dfrac{e^{y_i}}{\sum_j e^{y_i}}, & \text{if } i = c \\ \dfrac{e^{y_i}}{\sum_j e^{y_i}}, & \text{else} \end{cases} \tag{3}$$

显然, 只有 $i=c$ 时, $g_i<0$. 因此, 根据梯度 ∇W_L^i 各个维度的符号可以推测出真实标签所在维度, 攻击模型收敛更快、准确性更高. 以上的攻击关键是使得生成样本的梯度与目标样本的梯度足够相似. 通常的方式是采用欧式距离衡量二者相似度, Jonas 等人[27]则通过实验证明: 以梯度的方向为依据, 采用余弦相似度衡量梯度间相似性, 在高维参数下能取得更高的攻击准确率. 重建攻击通常发生于联邦学习训练阶段, 但是当预测阶段也需要多方协作或投票时, He 等人[28]采用相同的策略使攻击者同样可以根据终端的预测结果(prediction)重建测试样本. 但是当目标函数不连续、不可微时, 以梯度为攻击对象的样本重建方式将会失效. Pan 等人[29]针对此情形, 在几乎不需要攻击者背景知识假设的条件下, 提出了以嵌入式表示(embedding)为攻击对象的算法, 重建目标文本的模式(pattern)或关键词. 深度学习中, 如果输入样本是高维稀疏向量, 需要首先通过神经网络转换到低维向量空间中形成嵌入表示. 具体攻击流程如下: 算法假定攻击者知道终端生成嵌入式表示所采用的预训练语言模型, 在此基础上, 攻击者将公开数据集及其标签输入预训练模型生成嵌入式表示, 以嵌入式表示和对应标签作为训练集训练攻击模型, 用于预测目标嵌入式表示的所属类别. 具体攻击场景有两种.

- 模式重建攻击针对有固定模式(如基因序列)的文本, 攻击者尝试根据嵌入式表示恢复文本中的模式. 考虑到攻击模型的输出是敏感词汇, 大量词汇将导致模型更加庞大, 作者采用分治法将攻击模型拆分成一系列子攻击. 例如, 攻击用户的生日可以分为年、月、日这 3 个子攻击, 则总参数量将由 $O(|V(w_1)\cdot V(w_2)\cdot V(w_3)|)$降低为 $O(|V(w_1)+V(w_2)+V(w_3)|)$.
- 而关键词重建攻击中, 由于公开数据集与终端数据集的关键词差异, 可能导致领域迁移时不匹配、攻击准确性低. 为此, 作者采用对抗学习的思想, 训练模型学习统一的领域不变(domain-invariant)隐式表示, 在使不同领域文本的编码尽可能相似的同时, 完成敏感词重建任务. 该方法首次将嵌入式表示中的丰富信息还原, 为重建攻击提供了新的角度.

样本重建攻击不改变全局模型的正常训练, 且不需要额外构建新的模型, 恢复出的样本的信息更加丰富. 已有研究[27]证明, 任何全连接的网络结构都会受到样本重建攻击. 但是对于泛化良好、经过梯度压缩处理的模型, 样本重建攻击准确性大大降低.

2.2.2 推断攻击

攻击者根据模型的中间参数或预测结果, 推测给定用户的某条记录或某个属性是否属于该模型的训练集, 称为推断攻击(inference attack). 在第 2.2.1 节的重建攻击中, 隐私被直接定义为参与训练用户的数据, 攻击针对的是用户的全部特征或者某个敏感属性. 而 Shokri 等人[22]在集中式训练场景下首次提出的成员推断攻击, 为机器学习模型的隐私赋予了新的定义: 用户是否参与训练. 这样的定义来自现实应用: 如果可以确定目标用户参与了某疾病研究的训练, 那么该用户患病的信息就会因此泄露. 按照攻击者角色, 推断攻击分为外部和内部攻击: 外部推断攻击发生在模型训练完成并发布后, 由外部使用者发起; 内部推断攻击发生在模型训练阶段, 由内部参与方发起.

外部推断攻击中, 攻击者根据模型的预测结果推断给定记录的归属[5]. Shokri 等人[22]首次设计了黑盒情况下的外部成员推断. 算法首先生成近似样本训练影子模型, 然后以影子模型的预测值为特征、近似样本是否属于训练集为标签训练攻击分类器, 根据样本在目标模型中的预测值判断该样本是否属于目标模型的训练

集. 该攻击主要利用了模型的过拟合现象, 即模型在训练集和测试集预测准确度差异越大, 攻击的成功率就越高. 然而, 外部攻击者仅能依靠输出的置信度差异进行成员推断, 在泛化能力较好的模型上攻击能力下降.

目前, 有效的外部推断攻击仅能根据输出的置信度完成推断, 但是外部攻击者一般难以获取训练集的近似样本, 实际攻击能力有限. 相比之下, 内部推断攻击可依据的攻击目标非常丰富, 梯度、样本的嵌入式表示等所携带的信息量都高于输出结果的置信度. 通过衡量中间参数的差异、中间参数的分布和变化过程, 内部攻击者不仅能够推断出目标成员是否存在, 还能推断出目标用户的数据中是否含有目标敏感标签.

内部成员推断中, 攻击者能够判断给定的目标用户是否参与过模型训练. Melis 等人[30]针对嵌入层(embedding layer)中的非零向量, 首次设计了联邦学习中的成员推断. 以词向量为例, 显然, 只有在样本中出现的词语对应梯度才有更新, 而其他未出现的词语对应梯度为 0. 中心服务器在获取各个终端的梯度后, 即可判断给定的词语是否出现在目标终端的训练集中. 但是这种攻击所针对的隐私泄露是十分浅显的, 只能无序获取的词语集合, 对于词汇之间排列组合形成的复杂样本的推断能力不足. Nasr 等人[31]针对深度学习的梯度下降过程, 设计了更精细的成员推断攻击. 算法根据成员和非成员在模型上的梯度的区别进行推断. 半诚实的参与方通过掌握的中间参数重建目标模型, 采用类似自编码器(auto-encoder)的方式将中间参数编码后, 根据成员和非成员样本在模型上表现的差异完成推断. 而恶意的终端则能够进一步采用主动攻击方式, 将样本的梯度取反后, 更新模型并上传参数. 如果该参数能使中心服务器在后续更新的全局梯度大幅下降, 则是成员的可能性更高; 反之为非成员. 该攻击的成功来自重要假设: 梯度由于维度很大, 在深度学习中没有很好地泛化, 在成员和非成员上的范数分布不同. 由于模型参数的生成源于成员样本, 模型梯度也在成员样本上最大化. 因此, 模型在从未遇到过的非成员样本的损失函数上的梯度变化慢, 而在成员样本上的梯度下降快. 而如果某个样本的反向更新能使后续梯度通过大幅下降来“修正”这个“错误”更新, 则说明是该样本已经被模型学习过、其带来的损失值可以被模型修正, 因此是成员的可能性更高; 反之是非成员.

由上述算法可见, 降低输出精度、提高模型的泛化能力, 只能部分地提高模型的隐私性, 并不能完全抵御成员推断攻击. 这是由于隐私泄露并不仅仅来自过拟合, 也来自机器学习模型的记忆能力. 由于机器学习模型的复杂性, 训练样本在模型的结构类型以及参数上留下了痕迹, 使得模型参数和中间梯度中包含了大量的原始样本信息, 面对设计更精细的攻击时仍然存在风险.

内部属性推断中, 攻击者试图判断某个敏感属性是否存在于目标终端的训练集中. Melis 等人[30]设计的攻击能够推断与模型的主分类任务无关的敏感属性. 假设攻击者为中心服务器, 拥有的辅助集同时包含具有目标属性的样本和不具有目标属性的样本. 则攻击者能够根据全局参数和辅助集分别生成包含目标属性的梯度更新向量和不包含目标属性的梯度更新向量, 并以此为输入训练属性分类器, 判断观察到的梯度更新所基于的单个批量(single-batch)中是否含有目标属性. 在主动攻击方式下, 攻击者可以进一步修改模型的损失函数, 训练多任务分类器, 同时完成主分类任务和敏感属性推断任务. 属性能够被推断, 是由于训练过程中模型既学习与主任务相关的信息, 也学习到与算法任务无关的数据信息, 蕴含这些信息的模型中间参数导致了隐私的泄露. 而主动攻击中攻击者通过修改损失函数实现多任务学习, 进一步增强了模型对“不相关”信息即敏感属性的学习强度, 因而攻击效果更加突出.

2.3 隐私攻击分析与对比

仅仅通过中间参数, 内部攻击者就能提取大量隐私信息, 其原因在于中间参数与输入数据的联系十分紧密. 嵌入式表示通常是记录的低维向量表示, 每个维度都与某些维原始数据直接相关, 如果终端发送的是记录的嵌入式表示, 则有可能直接泄露信息; 如果终端发送的是梯度或模型参数, 同样有可能间接泄露信息, 且泄露的信息更加丰富. 这是因为梯度与特征成比例, 梯度的值能够反映本次迭代所用的训练数据的特征. 此外, 由于梯度及模型参数的维度很高不能得到充分泛化, 高维的梯度可能记住更多与模型训练目标无关的信息, 带来极大的隐私泄露风险.

内部重建攻击与内部推断攻击都通过窃取或篡改训练的中间参数, 获取与参与训练的原始数据相关的信息. 但二者的攻击目标不同: 重建攻击目标为某个终端数据, 而推断攻击的目标为所有终端的训练集信息.

我们将攻击依据的主要信息称为攻击对象. 重建攻击以本地参数为主要攻击对象, 通过训练生成对抗网络或者直接梯度下降的方式生成样本的表示; 推断攻击则以全局参数为攻击对象训练成员或属性分类器, 根据辅助集区分成员或属性是否存在. 而当攻击者没有辅助集时, 依然能够通过聚类算法或影子模型间接完成推断. 由于内部攻击者掌握的中间参数蕴含着丰富的信息, 半诚实参与方和恶意参与方都能对模型发起攻击. 恶意的参与方能够诱导攻击模型泄露更多信息, 使得攻击模型收敛更快、准确性更高.

外部重建攻击和外部成员推断攻击则通过获取公开的模型参数或输出获取与训练集相关的信息: 外部重建攻击的目标为部分训练集, 而外部成员推断的目标为训练集中的成员信息. 外部重建攻击对象包括模型的预测结果以及模型结构, 而外部推断攻击一般以模型的预测结果为对象发起攻击. 与内部攻击相比, 外部攻击可用的信息较少, 能够获取的攻击结果细节有限、准确性也更低.

与此同时, 重建攻击与推断攻击紧密相关. 样本重建能够为提供推断攻击更加丰富的背景知识. 当攻击者发起推断攻击但缺乏辅助集时需要借助影子模型, 而影子模型的所需要输入数据可以由样本重建攻击提供. 此外, 重建攻击中隐含着目标记录或属性存在于训练集中的假设, 因此, 重建攻击隐含着推断攻击的目的. 隐私攻击策略的对比见表 3, 其中, 实心圆表示“是”, 空心圆表示“否”.

表 3 隐私攻击策略

隐私攻击		重建			推断		
攻击者角色		外部	内部		外部	内部	
类型		训练集重建	类别重建	样本重建	成员推断	成员推断	属性推断
攻击目标		生成部分训练集的表示	生成训练集的类别表示	生成训练样本	判断记录是否存在于训练集	判断记录是否存在于训练集	判断属性是否存在于样本或训练集
攻击者可信程度	半诚实	●	●	●	●	●	●
	恶意	○	●	○	○	●	●
攻击对象	本地参数	○	●	●	○	●	○
	全局参数	○	●	○	○	●	●
	模型结构	●	○	○	○	○	○
	预测结果	●	○	○	●	○	○
背景知识		无或辅助集	无		无或者辅助集		辅助集

上述攻击策略的主要流程如图 2 和图 3 所示. 其中, 实线表示联邦学习的模型训练的过程, 虚线表示内部隐私攻击的过程; 带有数字的空心圆表示被动攻击的步骤, 带有数字的实心圆表示主动攻击的步骤; 带有字母的空心圆表示联邦学习训练的步骤.

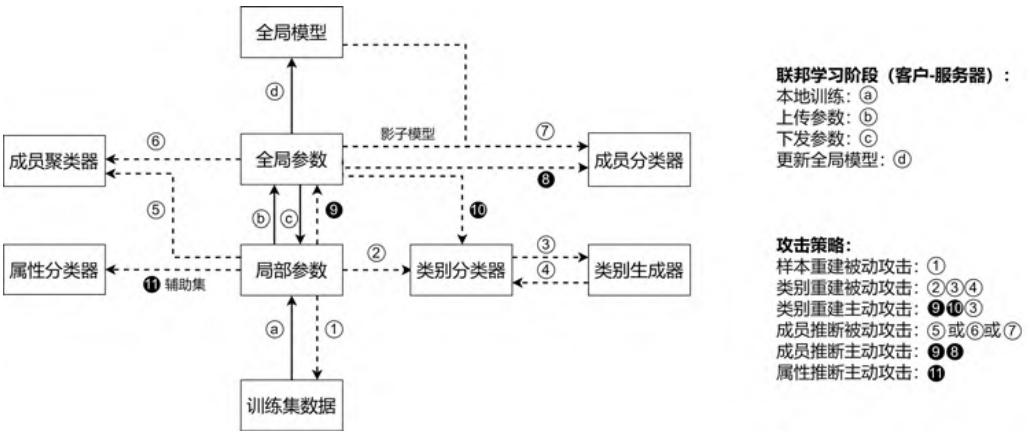

图 2 隐私攻击策略流程图(客户-服务器架构)

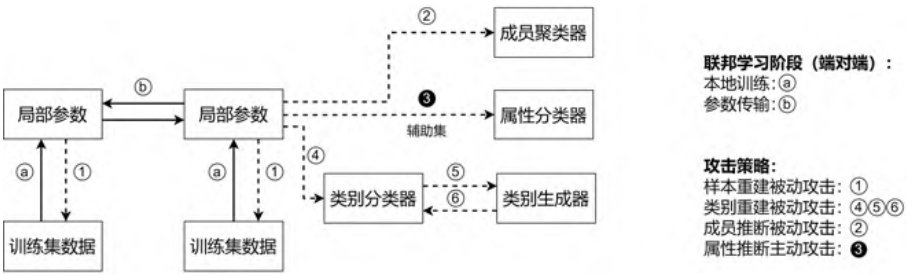

图 3 隐私攻击策略流程图(端对端架构)

一方面, 研究联邦学习的隐私风险和攻击策略有助于明确隐私保护的程度, 例如, Jagielski 等人[34]借助特定攻击衡量隐私算法的可能泄露程度, 从而计算隐私算法所需的最低隐私预算; 另一方面, 研究联邦学习的隐私风险和攻击也能明确隐私保护的对象. 联邦学习由于自身结构和训练过程的问题, 即使数据留在本地, 也依然面临着严重的隐私威胁. 保护联邦学习隐私除了需要保护发布后的模型以外, 还需要保护在更新过程中各个参与方上传、下发或者传输的中间参数. 而已有的集中式学习的隐私保护方法远远不足, 有效的联邦学习隐私保护手段是必须且亟需研究的领域.

3 隐私保护机制和技术

隐私保护技术是防御敏感信息泄露的技术, 能为信息的隐私提供严格的可量化的保护. 隐私保护的技术多种多样, 但总体分为两大方向: 信息模糊机制和过程加密机制. 信息模糊机制面向数据内容本身, 通过处理数据或参数使数据内容不易被关联到用户身份上; 过程加密机制面向数据传输的过程, 通过改变数据交换的形式使得传输过程中的数据不被识别. 两类机制使用的场景不同, 但都能在一定程度上抵御上述隐私攻击.

3.1 信息模糊

信息模糊机制的目的在于保护数据内容本身. 通过添加噪声等方式扰动原本特征清晰、极易识别的记录, 使得单条记录失去其独特性、隐藏在大量数据之中, 避免了记录来源泄露、记录所包含的敏感值的泄露. 同时, 经过精心设计的信息模糊能够使数据依然维持着原有的分布特点, 从而够支持模型的训练, 在一定程度上保证了数据的可用性. 信息模糊的手段包括中心化差分隐私、本地化差分隐私等技术.

3.1.1 中心化差分隐私

2006 年, Dwork 提出了差分隐私保护(differential privacy, DP)[35], 为隐私提供了一种定义: 群体的信息可以被公布, 但是特定的个体信息不能被泄露. 目标用户是否存在于训练数据集中, 对经过差分隐私保护的算法的结果几乎没有影响.

定义 1(差分隐私)[35]. 对于任意一个的算法 M, 令其输出的任意一个子集为 Ω. 若算法 M 在任意的相邻数据集 D 和 D' 上的输出都满足如下条件, 则称 M 提供(δ,ε)-差分隐私保护:

$$\Pr[M(D)\in\Omega]\leqslant\exp(\varepsilon)\cdot\Pr[M(D')\in\Omega]+\delta \tag{4}$$

其中, 相邻数据集指相差最多一条记录的两个数据集. 非负参数 ε 为隐私预算, 表示隐私保护程度. ε 越小, 隐私保护程度越高. 当 ε 趋近于 0 时, M 在相邻数据集 D 和 D' 上输出相同结果的概率分布趋同, 算法 M 可能泄露的信息也就越少. δ 也为非负参数, 表示 M 在 D 和 D' 上输出结果差异超过 $\exp(\varepsilon)$ 的概率, 即违背差分隐私保护的概率. 显然, δ 越小, 隐私保护程度越高.

特别地, 当 δ=0 时, 算法 M 提供最严格的 ε-差分隐私保护, 也被称为“纯差分隐私”保护.

上述差分隐私的定义在理论上保证了隐私, 实现则需要通过添加噪声的方式扰动数据. 具体实现机制包括拉普拉斯机制[35]、指数机制[36]、高斯机制[37]等.

定理 1(拉普拉斯机制)[35]. 对于任意一个函数 f, 定义 $M(D)=f(D)+Y$ 为拉普拉斯机制, 其中, Y 是从 $Lap(0,\Delta f/\varepsilon)$ 分布中抽取的独立随机变量, $Lap(0,\Delta f/\varepsilon)$ 表示均值为 0、方差为 $2\Delta f^2/\varepsilon^2$ 的拉普拉斯随机噪声. 拉普

拉斯机制 $M(D)$能够为算法 f 提供ε-差分隐私保护.

定理 2(指数机制)[36]. 对于任意一个算法 M, 如果选择元素 r 输出的概率正比于 $\exp\left(\frac{\varepsilon u(x,r)}{2\Delta u}\right)$, 则算法 M 满足ε-差分隐私. 其中, $u(x,r)$为评分函数, Δu 为评分函数敏感度.

上述两种机制主要用于中心化差分隐私的实现, 其中, 拉普拉斯机制针对连续的数值型数据, 指数机制保护离散的非数值型数据. 二者都借助全局敏感度设计噪声规模, 在聚合统计时, 需估计统计值边界和隐私泄露的边界.

3.1.2 本地化差分隐私

当收集数据的第三方不可信时, 中心化差分隐私就无法保护本地记录的隐私. 要使每一个记录或参数的隐私在本地得到保护, 需要本地化差分隐私技术(local differential privacy, LDP).

定义 2(本地化差分隐私)[38]. 给定 n 个用户, 对于任意一个算法 M, 如果在任意两条记录 t 和 t'输出相同的结果 t^*时满足下述条件, 则称 M 满足ε-本地化差分隐私保护:

$$\Pr[M(t)=t^*]\leqslant\exp(\varepsilon)\cdot\Pr[M(t')=t^*] \tag{5}$$

本地化差分隐私通过保证任意两条记录的输出相似性来保护用户隐私, 使得隐私保护的过程从数据收集方转移到用户本地, 避免数据收集中的泄露.

本地化差分隐私中, 除了可以采用常规的扰动机制之外, 更为普遍的是随机响应机制.

定理 3(随机响应机制)[39]. 主要用于本地化差分隐私的实现. 以问卷调查为例, 假设受访群体可以分为 A 和 B 两类, 为获得群体 A 的占比, 调查者发起查询群体类别的查询. N 个受访者, 每人以 p 的概率回答自己所在群体的真实情况, 以 $1-p$ 的概率回答与事实相反的情况. 若统计中有 n_1 个人回答“属于 A”, 有 $n-n_1$ 个人回答“属于 B”, 则可以获得群体 A 占比的无偏估计:

$$\hat{\pi}=\frac{p-1}{2p-1}+\frac{n_1}{(2p-1)n} \tag{6}$$

为保证其满足ε-本地化差分隐私保护, 隐私预算为$\varepsilon=\ln(p/p-1)$.

由于差分隐私具有后处理性[40], 参数经过本地化差分隐私保护后, 模型也将得到保护.

基于差分隐私技术实现的机器学习隐私保护方法主要包括输出扰动[41]、目标函数扰动[42]和梯度扰动[43]: 输出扰动[41]通过衡量函数的敏感度在训练完成的模型参数上添加噪声, 为整个模型提供差分隐私保护. 目标函数扰动[42]则在目标函数上添加扰动后训练, 模型可用性更高. 上述两种方式对目标函数有着诸多假设, 仅适用于经验风险最小化模型的中心化差分隐私保护. 梯度扰动[43]将梯度裁剪到一定范围后添加满足差分隐私的噪声, 隐私预算与训练迭代次数紧密相关. Yu 等人[44]发现: 梯度扰动过程中引入的噪声能够为凸优化算法提供一定的可用性保证, 从理论上证明了梯度扰动的有效性和算法优势. 梯度扰动对目标函数没有限制, 适用于大部分模型训练场景, 且本地化差分隐私技术和中心化差分隐私技术都适用于此方式.

差分隐私通过添加噪声将记录隐藏在整个数据集之中, 使得相邻数据集难以被分辨, 这与成员推断攻击的设定不谋而合. 因此, 差分隐私对于成员推断攻击有着很好的防御效果, 但是对于属性推断攻击、重建攻击的防御能力目前尚缺乏严格的证明. 此外, 尽管通过压缩隐私预算、增加噪声规模能够达到更高的隐私保护效果, 但是过多的噪声也会影响模型训练的准确性, 这是基于差分隐私的扰动方法不可避免的困境. 已有研究发现: 联邦学习上的差分隐私保护, 模型拟合度(fitness)与训练集大小的平方和隐私预算的平方成反比[45]. 因此, 在保持数据隐私的同时提高模型可用性, 差分隐私需要更加有适应性和精细的算法设计.

3.2 过程加密

过程加密的目的在于保护数据交换的过程. 通过密码学工具、安全多方计算等技术掩盖数据的交换过程, 只有特定的参与方才能获取协议指定的数据内容, 其余参与方在不破坏协议的情况下无法获取交换中的数据. 过程加密技术需要确保数据交换过程的安全性并控制计算代价. 能够保护隐私的过程加密的技术很多, 本文介绍两种联邦学习中常用的技术: 同态加密和秘密共享.

3.2.1 同态加密

同态加密(homomorphic encryption, HE)[46]提供了一种密码学解决方案, 使得终端可以对数据 x 加密后发送给第三方, 第三方能够在加密的数据上完成特定运算 f, 而不获取数据的明文信息. 终端收到加密运算结果后解密, 与在明文上的进行运算 f 的结果一致. 其加密机制包含如下过程:

$$\varepsilon=(KeyGen,Encrypt,Decrypt,Evaluate).$$

1) $(sk,pk)\leftarrow KeyGen(1^{\lambda},1^{\tau})$: 给定安全参数$\lambda$和参数$\tau$, 生成公钥和私钥的密钥对.
2) $c\leftarrow Encrypt(pk,b)$: 给定公钥和明文, 加密得密文.
3) $b\leftarrow Decrypt(sk,c)$: 给定私钥和密文, 解密得明文.
4) $\vec{c}'\leftarrow Evaluate(pk,\Pi,\vec{c})$: 给定公钥 pk、密文计算Π和密文向量 $\vec{c}=\langle c_1,...,c_t\rangle$, 在密文上完成运算并输出 $\vec{c}'$.

目前, 主要的同态加密方法都是基于误差还原问题(learning with errors, LWE), 即已知一个矩阵 A 和带有误差的乘积结果 $Ax+e$, 如何还原向量 x 的问题. 此外, 也包含整数分解问题、离散对数问题等传统方式. 根据这些困难问题, 可以实现部分同态加密[47]、全同态加密[48].

3.2.2 秘密共享

秘密共享(secret sharing, SS)[49]是安全多方计算中的重要机制, 在参与者共享秘密时, 保证信息不会被破坏、篡改或丢失. 秘密共享通过特定运算, 将秘密分成若干份额, 根据协议规定某些参与者可以联合恢复秘密. 以门限秘密共享方案为例: 协议将秘密 s 分为 w 个秘密碎片 s_i, 发送给 w 个用户保管. 只有 t 个以上的用户合作时才可以恢复秘密. 其形式化定义如下: 对于秘密分割函数 S, $S(s,t,w)\rightarrow\{\langle s_0\rangle,\langle s_1\rangle,\ldots,\langle s_w\rangle\}$, 存在秘密重构函数 R, $\forall m>t$, $R(\langle s_0\rangle,\langle s_1\rangle,\ldots,\langle s_m\rangle)\rightarrow s$.

目前的秘密共享方案主要包括 Shamir 方案[49]、Blakley 方案[50]、中国剩余定理方案[51]等.

在实际应用中, 同态加密技术在实际应用中还存在诸多障碍, 如计算效率低、存储开销大等性能问题. 此外, 尽管存在性能相对较好的部分同态加密算法(如 Paillier[52]等), 依然无法处理加法或者乘法之外的运算场景. 对于联邦学习模型训练中的激活函数等运算, 目前只能通过加法和乘法近似实现, 严重制约了计算精度. 而秘密共享技术虽然不需要大量计算, 却增加了通信次数, 加剧了联邦学习的通信负担. 但是两种技术都能满足掩盖数据内容、保护传输过程的需求.

上述的隐私保护机制和技术采用的原理不同, 有其各自适用的场景. 信息模糊机制通过添加噪声扰动数据提供保护, 形式灵活便于部署, 不增加计算和通信负担, 适用于大规模协同训练场景. 得益于差分隐私的后处理性, 发布后的数据可以得到持续的保护. 但是添加噪声会给模型带来一定的准确性损失, 其中本地化差分隐私由于提供的隐私性更高,模型的准确性损失更大. 相比之下, 过程加密机制借助同态加密、秘密共享等技术为数据的运算或发送过程加密, 能够在不损害模型准确性的前提下提供严格的隐私保护, 适用于参与方较少、对结果准确性要求高的训练场景. 但是由于密码学技术本身的限制, 同态加密计算量大、秘密共享通信量大, 协同训练的效率受到影响, 且对客户端在线与否有更高的要求. 由此可见, 隐私保护机制和技术各有长短, 研究者需要根据具体的模型、使用场景选择合适的技术、尝试结合不同的技术, 设计出高效、实用、严格的保护方案. 具体的隐私保护机制和技术对比见表 4.

表 4 隐私保护机制和技术对比

隐私保护机制	技术	原理	优点	缺点
信息模糊	中心化差分隐私	中心提供噪声, 扰动数据	计算效率高; 通信开销低; 后处理性, 保护发布后的数据;	一定的可用性损失; 要求中心服务器可信
	本地化差分隐私	本地提供噪声, 扰动数据	计算效率高; 通信开销低; 后处理性, 保护发布后的数据;	较大的可用性损失
过程加密	同态加密	数据加密, 在密文上运算	隐私保护严格	无法处理复杂运算; 计算效率低; 存储开销大
	秘密共享	数据分片, 多方协同方可安全聚合	计算效率较高; 隐私保护严格	计算效率低; 通信开销大

4 联邦学习中的隐私保护算法

基于上述隐私保护机制和技术, 学者们为联邦学习设计了多种保护措施. 尽管这些保护措施设置在训练的不同阶段, 但隐私保护的对象是明确且清晰的: 中心或本地. 中心是指中心服务器所掌握的中间参数和训练完成的模型; 本地则包括终端所掌握的数据和本地模型参数. 二者是联邦学习主要的隐私泄露位置. 因此, 本节以隐私保护的对象为线索, 将联邦学习隐私保护算法分为 3 种主要类型: 中心保护、本地保护、中心与本地同时保护策略. 中心保护策略以保护中心服务器所掌握的参数为目标, 考虑模型的使用者带来的威胁; 本地保护策略以保护本地所掌握的参数为目标, 考虑中心服务器带来的威胁; 中心和本地同时保护策略以保护所有参数为目标, 同时考虑模型使用者和中心服务器所带来的威胁.

3 种保护策略的区别如图 4 所示. 需要说明的是, 本地保护策略提供的保护有时也能起到防御模型使用者(外部攻击者)的效果, 但防御使用者并非本地保护策略的核心任务, 所以该防御范围在图中用虚线表示.

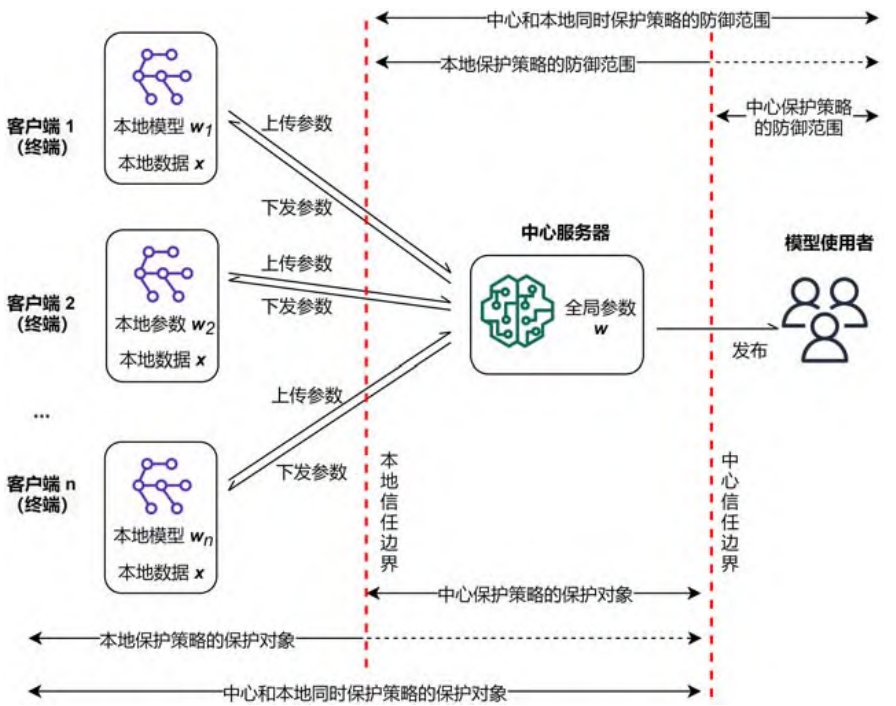

图 4 隐私保护策略的信任边界: 以客户-服务器架构为例

4.1 中心保护策略

中心保护策略针对客户-服务器架构, 关注联邦学习中心服务器面临的隐私泄露风险, 保护中心服务器获取和下发的全局中间参数, 并通过保护该参数间接为发布的模型提供隐私保护. 若不加保护地公开全局中间参数, 攻击者能够据此发起重建攻击或推断攻击; 公开训练完成的模型参数, 攻击者能够发起外部攻击, 均会导致本地数据泄露. 为降低模型被攻击的风险, 常用的保护方法有扰动和知识迁移.

4.1.1 扰 动

扰动(perturbation)方法是指通过差分隐私等技术, 在模型的训练过程中添加噪声扰动, 使得发布的模型在保持可用性的同时得到保护.

在集中式场景中, 隐私保护对象为单条记录. 扰动方法被用于确保攻击者无法准确地获知一条记录存在与否, 对于模型的输出影响较小. 但是针对一条记录的保护, 在联邦学习场景下是不够的. 例如, 在收集用户的输入记录训练语言模型时, 用户的敏感信息会出现在不止一条记录中且彼此相关. Melis 等人[30]已验证了记

录级别的保护无法抵御属性推断攻击. 为此, Geyer 等人[53]首次提出了在联邦学习下, 针对用户级别的差分隐私保护算法. 用户级别的隐私保护是指相邻数据集相差的是一个用户所有的记录, 其敏感度定义在相邻用户的数据集上. 具体来说, 在每次迭代中, 中心服务器随机选择 m_t 个用户下发全局平均梯度, 终端根据本地数据更新梯度Δw_k 后上传. 中心服务器为梯度聚合后添加满足差分隐私的高斯噪声 $N(0,\sigma^2S^2)$, 使得全局平均梯度受到(δ,ε)-差分隐私保护. 同时, 根据差分隐私的后处理特性, 训练完成的模型也将得到中心化的隐私保护. 为了计算隐私损失, Geyer 采用矩累计(moment account)[43]的方法获取隐私损失更紧的边界, 从而提高对隐私预算的利用率. 考虑到梯度无界, 难以预估的用户梯度范数范围会导致敏感度计算困难, Geyer 选择梯度的中位数 $S=median\{\Delta w^k\}$为阈值剪裁用户梯度, 使梯度敏感度被限制在有界范围$[-S,S]$之间, 以保障隐私预算. 对于第 t 轮迭代, 有如下更新:

$$w_{t+1}=w_t+\frac{1}{m_t}\left(\sum_{k=0}^{m_t}\Delta w^k\Big/\max\left(1,\frac{\|w^k\|_2}{S}\right)+N(0,\sigma^2S^2)\right) \tag{7}$$

但是, 一个用户的输入数据其实是一组向量, 每个敏感度不同, 直接剪裁的方式不仅增加训练迭代次数、降低模型准确性, 也浪费了一部分隐私预算. 并且Geyer没有对梯度的中位数做任何保护措施, 也存在一定的隐私泄露风险. McMahan 等人[54]进一步完善了用户级别的隐私保护, 实现了长短期记忆网络(long short-term memory, LSTM)的中心保护. 该方案通过设计逐层剪裁的方式为模型的每一层设计不同的剪裁阈值, 根据终端用户样本量加权平均计算总剪裁敏感度, 给拥有更大的样本量的用户以更高的权重. 然后据此设计隐私参数, 为梯度添加满足差分隐私的高斯噪声. 作者在随后的工作[55]中进一步拓展, 考虑了在小批量梯度下降中, 一次更新的输入是多个用户的情况. 该算法将各组向量分别缩放到同样的范数大小后添加噪声, 再将其还原. 缩放的优势在于快捷地计算一个批量敏感度, 并且整体考虑所有样本时, 辅助信息不会被忽略. 但是该算法的缩放因子需要预先设计, 这意味着每组向量的范数都需要预估, 实际应用中是比较困难的. Liu 等人[56]则在离散分布学习场景下, 为用户级别差分隐私保护给出了用户数量所需要满足的复杂度理论上界. 但是在梯度下降过程中引入噪声, 始终存在误差大、模型准确性低等问题. 为此,Liang 等人[57]结合拉普拉斯平滑机制[58]降低梯度方差, 提高梯度计算的准确性, 分别在均匀采样和泊松采样机制下降低噪声带来的负面影响, 提高了模型可用性.

上述方法中, 本地参数在上传时不添加任何保护, 噪声均由中心服务器添加, 参数的聚合结果满足中心化差分隐私定义. 但是中心服务器可能会忽略噪声的添加, 致使终端隐私泄露. 更为安全的方式是中心化的差分隐私在本地实现, 即在本地添加噪声, 使得聚合结果满足中心化差分隐私. Shokri 等人[59]提出了梯度本地扰动算法. 在每一轮迭代时, 终端随机选择大于预设阈值的梯度, 为其添加满足差分隐私的拉普拉斯噪声后上传, 中心服务器收集梯度后取平均值完成本次迭代. 该算法虽然为梯度上传和模型发布提供了隐私保护, 但是隐私保护的程度会随着迭代次数的增加而下降. 此外, 由于敏感度和阈值的设置过于宽松, 噪声对模型的准确性也产生了一定的影响. Wu 等人[60]借助输出扰动机制提高梯度扰动部署的便捷性, 利用序列的随机性收紧梯度的敏感度边界, 在同样的隐私预算下, 终端的参数添加的噪声更小, 模型的可用性提高. 除了常见的梯度下降方案以外, 研究者还为更多优化方案设计了适应性保护方案. Hu 等人[61]为优化双变量的损失函数, 本地为双变量的块坐标下降法(block coordinate descent, BCD)中交替更新的参数分别添加高斯噪声, 但是该方案仅支持光滑的目标函数. Huang 等人[62]则提出了交替方向乘子法(alternating direction method of multipliers, ADMM)的扰动算法, 为非光滑的凸目标函数提供了可用性较高的保护方案. 该算法不追求每轮迭代中增广拉格朗日函数所带来的过高精度, 转而采用带有 L_2 范数项的目标函数一阶近似作为优化目标. 近似函数光滑的特点, 不仅简化了求解过程、为后续差分隐私提供了有界的目标函数的 L_2 敏感度, 还免除了以往方案对原目标函数的光滑、强凸的假设. 为提高算法的稳定性和收敛性, 终端在为更新的参数添加高斯噪声时, 还能确保高斯噪声的方差随迭代次数增加而减少, 在训练后期逐步降低噪声对训练的影响. 该算法的隐私预算由矩累计[43]计算得出, 受到差分隐私保护.

上述方案中, 参数的聚合过程满足中心化差分隐私, 但是在下发时, 由于全局参数求和取均值, 导致噪声

被稀释, 无法再为下发的参数提供相同的保护水平. Wei 等人[63]设计了分阶段差分隐私保护模型 NbAFL. 作者首先将参数分为由本地上传和由中心下发两种. 在训练开始前, 服务器向所有 N 个终端广播隐私参数(δ,ε)和模型初始化参数 w_0. 在本地模型训练完成、参数上传的阶段, 各个终端为自己的更新参数添加满足中心化差分隐私保护的噪声. 为此, Wei 等人将各终端的敏感度定义在终端相邻数据集 D_i 和 D_i' 上, 并将模型参数范数剪裁至范围$[-C,C]$之内, 得到的终端 u_i 敏感度为 $\Delta_{S_u}^{D_i}=2C/|D_i|$. 令 m 为所有终端数据集中的最小规模, 全局敏感度为 $\Delta_{S_u}=2C/m$. 若每个终端仅上传 L 次, 则对应的噪声规模为 $\sigma_U=cL\Delta_{S_u}/\varepsilon$. 此时, 参数上传聚合后的结果满足中心化差分隐私. 在服务器完成模型平均、参数下发的阶段, 服务器为更新参数添加高斯噪声. 该算法将敏感度定义在相邻参数 w 和 w'上, 同样将参数剪裁后得到敏感度 $\Delta_{S_d}=2C/mN$. 为保证 T 次下发参数得到中心化差分隐私的保护, 添加的噪声规模须达到 $\sigma_A=cT\Delta_{S_d}/\varepsilon$. 作者充分利用上传阶段所添加的噪声, 在已有的扰动参数结果上仅添加 $\sqrt{\sigma_A^2-\overline{\sigma_U^2}}$ 大小的高斯噪声. 该方法全面地考虑了中心参数不同阶段的隐私, 精细地设计噪声, 但是依然需要多轮迭代才能到达较高的模型准确率.

4.1.2 知识迁移

终端训练的模型接触了本地的敏感数据, 直接使用其参数或者模型存在极大的隐私泄露风险. 一种思路是采用知识迁移(knowledge transfer)的思想[64], 以终端的敏感模型为教师, 中心服务器学习其预测能力训练新的学生模型. 仅发布学生模型, 终端的泄露训练数据的风险大大降低.

Hamm 等人[65]首次提出了基于迁移学习的多方隐私保护算法. 该算法中, 中心服务器从 M 个终端收集本地模型作为教师标注辅助的非敏感的公开数据集, 然后统计所有教师的投票结果作为公开数据集的最终标签. 中心服务器利用标注完成的公开数据集重新训练学生模型, 并添加输出扰动机制[41]使得训练完成的模型受到差分隐私保护. 然而, 直接采用“少数服从多数”的投票方式会带来过高的敏感度, 造成严重的隐私损失. 为此, 作者在投票时设计了加权求和的方式, 使得投票由确定的结果变为概率, 不仅使敏感度降低了 M 倍, 还提高了模型的稳定性. 但是该算法对模型的准确性影响较大, 并且由于输出扰动机制的限制, 只能用于凸损失函数的模型训练.

Papernot 等人[66]在此工作的基础上提出了 PATE 模型, 直接在知识迁移的中间过程中添加保护, 进一步保护隐私. 该模型框架如图 5 所示, 算法同样以终端模型为教师标注辅助的公开数据集, 但是中心服务器仅收集教师的投票结果作为最终标签. 为保护原始数据集, 投票结果在添加满足差分隐私的拉普拉斯噪声后用于学生模型的训练. 由于差分隐私的后处理性, 学生模型也满足差分隐私. 与 Hamm 等人的算法[65]相比, PATE 中学生模型仅能接触到扰动后的最高投票结果, 隐私性提高, 并且支持更加丰富的机器学习模型训练. 作者在随后的工作[67]中对模型改进, 将添加的噪声换成高斯噪声, 并采用雷尼差分隐私(Rényi differential privacy, RDP)[68]进行保护. 同时提出了基于置信度和模型交互两种机制, 隐私预算得以被充分利用. PATE 的可用性来自大量教师模型的投票, 因此难以处理复杂任务. Sun 等人[69]设计了不变最大噪声机制, 使得输出更加稳定从而能够容忍更多噪声. 由此, 学生模型向教师模型发出请求次数的上限得到提高, 提高了训练的准确性.

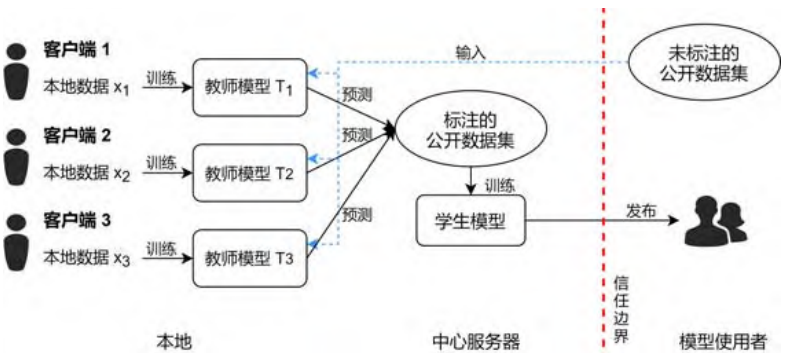

图 5 教师隐私聚合模型(PATE)架构

知识迁移方法的优势在于发布的模型不直接接触敏感数据, 而由投票机制形成的聚合结果既隐藏了接触过敏感信息的教师模型, 又提供了相对稳定的输出结果.

4.2 本地保护策略

本地保护策略适用于联邦学习的所有架构. 该策略关注终端面临的隐私泄露风险, 保护终端向中心服务器上传的参数或向其他终端传输的参数. 终端的真实参数一旦被其他攻击者获取, 攻击者能够据此发起内部重建攻击, 导致参与训练的数据泄露. 因此, 在终端公开参数之前添加隐私保护而不是盲目信任第三方, 是一种安全、有效的做法. 常用的保护方法包括扰动和加密.

4.2.1 扰 动

本地化扰动(local perturbation)是指通过差分隐私等技术扰动终端模型的训练过程、保护本地的参数、降低其被攻击的风险的方法.

在端对端架构中, 终端直接发布经过中心化差分隐私保护的模型, 能够为本地数据提供严格的隐私保护. Zhao 等人[70]提出了基于回归树的差分隐私保护方法, 并利用梯度提升决策树(gradient boosting decision tree, GBDT)中的提升(boosting)特性, 设计了模型的隐私聚合算法. 该系统由多个终端构成, 共同训练 GBDT 模型. 由首位终端 u_1 根据本地数据构造一棵决策树, 模型参数发送给下一个终端 u_2; u_2 根据模型和本地数据计算残差, 拟合回归树, 并将模型参数发送给下一个终端 u_3. 循环该过程, 直至所有的终端训练完成. 为了保护每个终端的模型参数, 作者设计了两阶段的差分隐私保护方式: 在构造树时, 终端在树的每一层采用指数机制选择最大信息增益对应的特征作为最佳分割点, 使本地模型满足ε_1-差分隐私; 在利用残差训练提升回归树时, 终端在损失函数的分子和分母上分别添加拉普拉斯噪声, 使得本地模型满足ε_2-差分隐私. 在各终端拥有的数据不相交的前提下, 最终的模型将满足$(\varepsilon_1+\varepsilon_2)$-差分隐私. 该算法的隐私预算大小仅与树的高度相关, 不依赖数据维度和迭代轮数, 因此即使隐私预算较小, 模型也能达到较高的准确性. 但是该算法仅适用于提升机制, 难以在一般的模型训练中推广. Ding 等人[71]基于 ADMM 框架提出了更加通用的端对端参数更新和保护方案, 进一步节约了隐私预算. 在本地训练阶段, 终端首先收集来自其余终端广播的参数更新本地模型, 然后将部分隐私预算用于扰动目标函数, 部分用于扰动更新的参数, 使本轮迭代满足ε_1-差分隐私保护. 在参数传输阶段, 终端采用稀疏向量技术(sparse vector technique, SVT)检查本地参数与上一轮全局参数的差距, 如果小于预设阈值, 就放弃将本地更新广播给其他终端, 避免隐私预算的浪费. 而此阈值同样受到ε_2-差分隐私保护, 其敏感度通过剪裁损失函数后, 计算两次更新的距离得到. 最终, 作者采用 zCDP 技术(zero-concentrated differential privacy, zCDP)[72]将差分隐私的全部预算收紧到$(\varepsilon_1+\varepsilon_2)^2/2$, 进一步降低隐私开销, 提高模型的可用性. 采用差分隐私的变体定义设计本地扰动方法来改善模型的可用性或隐私性, 对于纵向联邦学习同样可行. Wang 等人[73]借助 JDP[74]技术同时保护纵向联邦学习两方的中间参数和模型权重, 提高模型训练和发布的隐私性. Lu 等人[75]则通过评估本地参数对全局模型的贡献, 实现了隐私预算的动态分配. 在每一轮迭代中, 客户端收集其他终端对全局模型的验证得分和扰动参数, 并以此为权重加权求和更新全局模型. 同时, 客户端使用自己的数据集验证全局模型的绝对平均误差, 并据此计算验证参数的质量得分. 客户端通过为得分高的模型参数分配更多隐私预算, 实现全局隐私预算分配. 该方案通过为质量更高的参数添加更少的噪声, 提高模型训练的准确性.

在客户-服务器架构中, 终端发送经过本地化差分隐私技术保护的参数, 能够为本地数据提供隐私保护. 然而, 传统的本地化差分隐私技术仅针对一个数值, 直接将本地化差分隐私保护技术应用于机器学习中高维梯度的扰动, 将带来过大的噪声干扰, 使得模型准确性严重下降. 为适应联邦学习高维参数的扰动需求, 研究者以经典的本地化差分隐私技术为基础, 设计了灵活的隐私预算分配机制和优化的扰动输出值, 以提高模型的准确性. Wang 等人[76]提出了针对高维数据的本地化差分隐私技术. 通过结合拉普拉斯机制变体[77]和 Duchi 的有界噪声机制[78]的优点, Wang 等人设计的 PM 机制既保证了输出的扰动值有界, 又保证了为每个数值添加了最优的独立噪声. 考虑到高维度的影响, 终端进一步在全部的 d 个属性中随机选择 k 个维度上传参数, 则使得噪声的均值误差降低 $\sqrt{d}$ 倍. 在隐私预算较大时, 该算法的准确率有大幅度提升. 但是其估计误差依赖于模

型迭代的轮数, 迭代轮数的增多会导致算法可用性的下降. Arachchige 等人[79]采用输入扰动的思想设计了基于本地化差分隐私的 LATENT 算法, 为神经卷积网络设计扰动机制. 网络的卷积和池化模块部署在终端用于抽取特征, 终端将特征转为二进制编码后随机扰动, 然后发送给中心服务器. 由于编码后的特征维度过高, 采用一元编码扰动技术[80]带来的过高的敏感度将导致过多扰动, 破坏原始数据的分布. 作者在一元编码扰动的基础上改进, 通过设置参数α, 灵活控制编码 0 被扰动为 1 的概率. 同时, 结合 OUE 算法[81]分别设计扰动 0 和 1 的概率, 通过牺牲编码 1 的稳定性来保护编码 0 的输出稳定性, 使整个编码向量保持稀疏, 保证扰动后的特征的准确性. 该算法仅需要终端上传一次扰动特征中心服务器就可以网络的训练, 因此隐私预算与迭代次数无关, 准确率较高. 但由于发送的是本地特征向量, 对于数据量较大的训练场景, 通信代价将大大升高. 以往的本地化差分隐私技术难以兼顾不同隐私预算下的数据估计的准确性, 为此, Zhao 等人[82]针对预算较小的情况提出了 3 种不同概率输出扰动值的本地化差分隐私技术, 在预算较大时则提出优化的 PM 机制, 兼顾不同预算的同时提高模型准确性. Truex 等人[83]在扰动本地参数时引入α-CLDP[84]方法(α-condensed local differential privacy, α-CLDP), 根据输入样本对的距离分配隐私预算, 以较大的概率输出与原始值相近的扰动值, 以较小的概率输出相距较远的扰动值. 该方案进一步减少扰动为数据分布带来的影响, 在相同的隐私预算下, 模型准确性高于中心化差分隐私保护下的联邦学习模型. 上述方案中的扰动对象均为连续型数值, Wang 等人[85]以文档主题生成模型(latent Dirichlet allocation, LDA)为背景, 为生成模型中的离散值扰动提供了优化的本地保护算法. 由于 LDA 模型候选词数量庞大, 以其作为候选值进行扰动不仅带来巨大的隐私预算消耗, 还会引入过多噪声降低模型准确性. 因此, 传统针对离散值扰动的 kRR[86]本地化差分隐私算法并不适用于 LDA 模型的训练. 改进方案根据实际候选词在整个文档中的权重筛选出重要的候选词列表, 作为候选值进行扰动, 通过消除扰动算法与文档词典大小的相关性, 降低噪声的影响, 从而提高模型训练效果.

梯度或者模型参数的维度通常较高, 直接发送本地扰动参数将带来巨大的通信量, 并且扰动高维参数也会引入大量的噪声, 影响模型的准确度. 因此, 在保护隐私的同时降低通信代价、提高训练效率, 成为研究者关注的焦点. 最直观的方法是仅选择部分参数发送. Shokri 等人[59]最早在中心保护策略中提出了随机选择发送的方案, 终端选择大于预设阈值的梯度发送, 默认低于阈值的梯度为 0. 为了避免预设阈值泄露隐私, 作者为阈值添加拉普拉斯噪声, 使之满足差分隐私. Truex 等人[83]则通过中心服务器随机选择预设数量的终端参数, 直接降低通信代价和隐私消耗. 然而, 预设值不能良好地适应实际训练数据, Liu 等人[87]提出了改进的发送方案. 本地在上传参数前, 依据指数机制选择权重最高的 k 个维度, 然后扰动选择的梯度元素, 其选择和扰动过程分别满足差分隐私. 梯度通常比较稀疏, 权重最高的梯度维度包含着主要信息, 因此仅选择部分梯度上传对模型可用性的损失有限, 当 k 较小时, 通信量大大降低. Shin 等人[88]采用随机投影对梯度先降维再扰动, 由中心服务器作变换恢复梯度, 但准确率有待提高. 上述方法针对梯度的数量或维度总数作出选择和压缩, 而 Agarwal 等人[89]首次提出了针对通信效率的隐私保护算法 cpSGD 算法, 压缩了梯度的每个维度. 针对梯度的每一维, 采用 k 阶离散技术[90]将数值离散到第 j 阶区间内, 其中, $j\in\{0,1,\ldots,k-1\}$, 然后以二进制字符串表示区间 j, 然后添加二项分布噪声保护隐私. cpSGD 将每一维数值所占的比特位数由 $O(\log(nd))$降至 $O(\log(\log(nd)/\delta))$. 类似地, Zhao 等人[82]采用离散后处理技术将扰动后的值离散到偶数个区间$[-C,C]$之内, 仅需要 2 位即可表示一个输出值. 这类方案提供了维度级别的扰动压缩方案, 节约了通信开销, 但将数值离散到区间的做法, 牺牲了模型一定的准确度.

4.2.2 加 密

加密(encryption)方法通过采用密码学工具为数据的传输过程提供强有力的隐私保证, 同时不破坏数据原有的值和分布. 常见的本地加密方法包括同态加密和安全聚合.

Aono 等人[91]采用加法同态加密机制为客户-服务器架构的联邦训练提供保护. 在训练开始前, 由任意一个终端生成密钥对并分享给其他终端. 训练开始后, 各个终端用公钥对本地的梯度加密上传. 中心服务器直接对密文计算实现加法同态后下发, 终端利用私钥解密获得更新的全局梯度. 而全局模型的初始化只需任意一个终端将初次训练所得的模型参数加密上传, 再由中心服务器分发给所有终端即可. 完整的系统架构如图

6 所示. 该算法是联邦学习通用的同态加密方案, 适用于 LWE, Paillier 等多种非对称同态加密机制. 但是由于所有终端都掌握着相同私钥, 一旦某个终端窃取其他终端发送的密文, 则能够立刻解密获取明文信息. 更加值得注意的是, 该算法仅关注本地参数的隐私性, 全局梯度对于所有终端都是直接可见的, 存在被攻击的可能. 因此, 由可信的第三方掌握私钥, 终端仅掌握公钥, 是另一种常见的同态加密方案. Yang 等人[11]在纵向联邦学习中的线性回归模型实现了基于加法同态加密的隐私保护: 首先, 由不掌握标签的被动终端 A 将与 A 相关的中间层结果和损失函数结果加密后, 发送给掌握标签的主动终端 B. B 据此求得完整的损失函数和模型中间层结果后, 加密发送给 A, 并将完整的损失函数的计算结果加密发送给可信第三方 CSP. 此时, A 与 B 完成了参数的交换, 得以分别在密文上计算各自的梯度, 然后根据 CSP 解密的损失函数和梯度更新本地模型参数. 但是损失函数的计算结果依然被暴露给了 CSP. 类似地, Liu 等人[92]也为迁移联邦学习设计了中间参数的加密方案. 上述两种算法的保护对象虽然是纵向或迁移联邦学习, 但是所采用的方法与文献[91]思路相近, 都属于本地策略中的同态加密方法. 这类方案基于同态加密, 同时保持了终端参数的隐私性和精度. 但由于同态加密技术的计算代价十分昂贵, 在实践中并不适用于大规模、大数据参与的模型迭代训练.

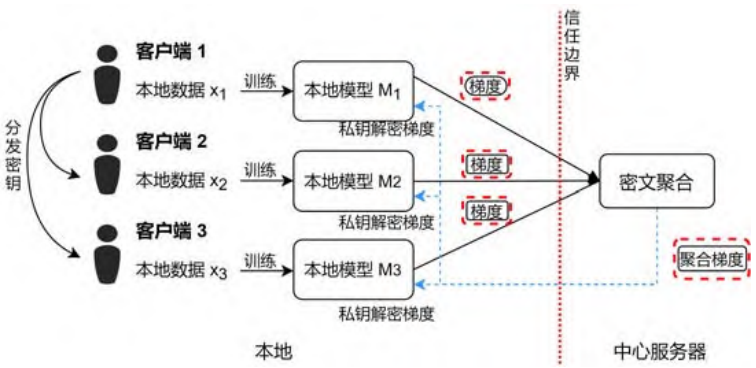

图 6 梯度同态加密系统架构

为了避开同态加密技术带来的过高计算代价, Google 团队提出了在一种客户-服务器架构下高效的安全聚合方案[93], 使得终端上传参数后, 服务器仅能掌握聚合后的结果, 而无法获知各个终端的真实值. 安全聚合对中心服务器的可信度要求低. 通过 Diffie-Hellman(DH)密钥交换, 任意两个终端 u_i 和 u_j 之间共享一个随机数 s_{ij}, 约定分别在参数上增/减 s_{ij} 掩盖真实值, 只有在求和之后才能抵消. 其掩盖函数如下:

$$y_i = x_i + \sum_{j \in U: i<j} s_{i,j} - \sum_{j \in U: i>j} s_{j,i} \pmod R \tag{8}$$

考虑到实际情况中, 终端掉线将导致随机数无法抵消、聚合失败, 终端需将 DH 种子用秘密分享技术分享给其他用户, 确保至少 t 个终端在线, 服务器就可以恢复种子获取没有抵消的随机数. 但是对于因网络延迟而迟到的用户, 这样的恢复方式会使其数据直接暴露. 为此, 作者提出了双层掩盖(double-mask)的方式, 为终端引入一个新的随机数 b_i, 将其秘密分享给其他终端. 在秘密恢复阶段, 对于掉线终端, 其他终端仅上传 s_{ij}, 对于在线终端, 其他终端仅上传 b_i, 确保中心服务器既能正确聚合, 又可以一次性解决客户端延迟和掉线问题. 完整的掩盖函数如下:

$$y_i = x_i + PRG(b_i) + \sum_{j \in U: i<j} PRG(s_{i,j}) - PRG\left(\sum_{j \in U: i>j} s_{j,i}\right) \pmod R \tag{9}$$

其中, PRG 为伪随机数生成器.

上述的安全聚合模型为本地参数的发送提供了严谨的隐私保护, 但是终端之间能够相互通信的假设, 在实际应用中并不具有一般性. Heikkilä 等人[94]设计了由多个服务器组成的安全聚合方案. 终端将自己的参数随机分片后, 加密分别发送给多个中心服务器, 任何一个服务器都无法获知参数碎片对应真实数据, 仅能在收集完所有用户的参数碎片后共同聚合并解密, 获得聚合结果. 此方案中, 本地的参数不会被泄露, 也不需要与

其他终端通信协作完成. Zhang 等人[95]结合了安全聚合和同态加密方法, 由中心服务器聚合上传的梯度后解密. 为防止本地梯度泄露, 采用秘密分享技术确保至少 t 个用户上传参数之后, 服务器才能解密. Liu 等人[92]采取同样的思路, 设计了迁移联邦学习的保护方案. 作者借助秘密共享技术为训练过程中需要交换的中间参数提供保护. 安全聚合的方案基于密钥协商和秘密分享等密码学工具, 与同态加密方法相比, 计算代价大大降低, 但是增加了通信的次数, 其通信代价和计算开销也高于扰动方法. 在安全聚合的基础上, Xu 等人[96]为进一步验证终端收到的参数的完整性, 采用同态哈希函数和伪随机技术验证中心服务器下发的参数.

4.3 中心与本地同时保护策略

中心和本地同时保护策略适用于联邦学习的客户-服务器架构. 该策略同时关注中心处和本地处的数据隐私, 具体包括终端上传的参数和中心服务器下发的参数、发布的模型. 只有中心和本地同时得到保护, 联邦学习才能抵御内部和外部攻击. 由于该策略涉及多个训练阶段, 往往需要借助扰动、安全多方计算、安全混洗等多种技术协同为训练过程中的数据提供保护. 如何有效结合多种隐私保护技术降低隐私损失、保持模型质量, 是中心和本地同时保护策略面临的关键问题. 合适的保护策略能够以较低的代价抵御内部和外部攻击, 形成对联邦学习从本地数据训练到模型发布全阶段的完整保护.

4.3.1 中心扰动与本地扰动结合

采用本地扰动方法虽然能够保护本地参数的隐私, 并且为之后的训练的模型提供后续(post-processing)的隐私保护, 但是对数据可用性的影响较为严重. 而中心扰动方法则依赖于服务器完全可信的假设, 在很多场景下并不实用. 中心与本地共同扰动, 为模型的本地和中心都提供了所需的隐私保护.

Avent 等人[97]提出了混合差分隐私保护模型, 既兼顾模型的可用性, 又满足不同用户的隐私需求. 终端根据个人意愿被分为自愿用户和一般用户. 中心服务器为自愿用户的数据提供中心差分隐私保护, 而一般用户在经过本地化差分隐私保护后才上传数据至中心服务器. 模型提供了统计最频繁查询-应答对的隐私保护模型: 首先, 中心服务器在满足中心差分隐私的算法下统计生成的候选高频查询对, 并估计概率和方差; 一般用户根据候选集分别为本地的查询和应答添加满足差分隐私的噪声, 中心服务器根据扰动后的数据估计一般用户的概率和方差; 最后根据两类用户的方差, 对最频繁对的概率加权求和. 由于自愿用户的扰动候选集合可以发布, 一般用户免于重复计算, 既节约了隐私预算, 又提高了模型的可用性. 并且两种用户的统计误差来源不同, 混合也提高了统计的准确性.

Zhao 等人[98]则在本地和中心分别设计了隐私保护算法. 终端通过目标扰动机制保护本地隐私. 训练模型时, 终端采用函数机制(functional mechanism, FM), 根据将复杂的目标函数泰勒展开, 取有限项近似表示以便计算函数敏感度, 从而为近似目标函数添加满足ε_1-差分隐私的拉普拉斯噪声. 根据差分隐私的后处理性, 其训练所得模型参数也得到差分隐私保护. 而中心服务器获取参数后, 采用指数机制, 选择k个终端的模型参数w_t聚合平均. 指数机制的效用函数则根据辅助集在各个本地模型参数 w_t上的准确度计算. 该步骤使得全局参数得到ε_2-差分隐私保护, 整个联邦学习训练和模型结果受到 $\max(\varepsilon_1,\varepsilon_2)$-差分隐私保护. 中心服务器根据辅助集计算各个本地模型准确度, 能够有效地识别效果过差的终端, 避免全局模型的可用性被某个不可靠的终端拉低. 但是采用辅助集的做法一来假设过强, 无法在实际中通用; 二来以辅助集为准训练出的全局模型, 其能力也会近似由辅助集训练的模型, 无法实现联邦学习综合所有终端信息的目的, 联邦学习的意义被大大削减. 但该方法同时考虑本地和中心的隐私, 为模型提供了严格的保护.

4.3.2 安全计算与扰动结合

安全多方计算技术(secure multi-party computation, SMC)需要密码学工具支持, 在无可信第三方的情况下构造协议, 根据各方的秘密输入共同计算函数, 完成训练. 尽管安全多方计算能够保护各个参与方的参数交换过程, 但单纯的安全多方计算依赖于大量密文计算、安全证明, 严重制约计算效率, 且无法保护模型发布阶段的隐私. 因此, 许多研究将安全计算与扰动相结合, 分别保护本地和中心的隐私.

Pathak 等人[99]首次提出了安全多方计算和扰动在多个参与方协同训练模型时的结合应用. 该模型通过安全多方计算的方式获取最小的数据集大小, 然后根据输出扰动机制[42]为全局模型参数添加噪声, 使发布模型

受到差分隐私保护. 该模型的训练分为 3 个阶段.

(1) 中心服务器获得数据集最小的终端的扰动下标 $\tilde{j}$: 设共有 N 个终端, 每个终端 u_i 将自己的数据集大小 n_i 分为任意两个值, 通过安全多方计算技术, 使中心服务器获取最小数据集的乱序脚标 $\tilde{j}$.

(2) 中心服务器和终端合作生成满足差分隐私的噪声: 各个终端 u_i 根据本地数据集大小生成第 i 维噪声, 服务器确保仅仅第 $\tilde{j}$ 维噪声(即真正掌握最小数据集的终端所产生的噪声 η_j)可用.

(3) 终端采用秘密共享技术发送噪声值并上传参数, 使得中心服务器仅能获得扰动后的聚合结果 $\sum w_i+\eta_j$. 由此, 模型得到中心化差分隐私的保护, 本地噪声 η_j、数据集最小的终端都没有在训练过程中暴露.

此外, 由于本方法基于输出扰动机制, 仅与数据集大小 n 相关而与训练的轮数无关, 隐私预算较低. 然而, 作者没有考虑到终端上传的参数泄露隐私的可能性, 没有对上传参数给予足够保护. Jayaraman 等人[100]在此基础上作出改进, 采用 zCDP 压缩梯度扰动带来的隐私损失, 降低了输出扰动的隐私预算.

上述方法虽然利用安全计算隐藏了本地扰动的噪声值, 但是由于协议复杂、输出扰动对目标函数的要求严格, 不具有通用性. Truex 等人[101]基于 Paillier 同态加密机制[102]设计了更为通用的方案, 该方案利用安全多方计算隐藏整个本地扰动参数, 借助秘密共享技术控制解密的时机, 确保聚合结果的明文在得到足够的噪声保护后才会被中心服务器获取. 在训练开始前, 终端基于门限变体机制[103]共享密钥 sk, 确保任何小于阈值 $n-t+1$ 个终端都不能解密. 训练开始后, n 个终端在为本地参数添加噪声后, 用公钥 pk 加密发送给服务器. 服务器借助 sk_i 解密全局参数的聚合明文, 完成本次迭代. 由于同态加密保护了各个终端上传的参数, 本地添加的噪声只需要满足中心化的差分隐私即可使全局参数得到保护. 相比于本地化差分隐私需要添加的噪声规模, 该方案在至少 t 个终端不共谋的前提下, 噪声标准差减少了 $\sqrt{t-1}$ 倍. 类似地, Xu 等人[104]借助函数加密机制[105]确保 t 个以上用户上传后, 中心服务器才能解密获取聚合结果的明文, 噪声减少量相同, 且通信代价进一步降低, 但是需要一个可信的第三方生成密钥. Kim 等人[106]借助可信第三方管理密钥, 服务器仅负责对加密参数运算, 解密和下发由可信第三方完成. Zhao 等人[107]则引入了半诚实的第三方代理服务器 P 协助完成协议, 同时降低中心服务器的通信负担. 在准备阶段, 中心服务器和每个终端都生成自己的公钥和私钥对, 并完成公钥交换. 在参数下发阶段, 中心服务器用 m 个终端的公钥为全局参数加密, 生成 m 份密文后发送给 P, 由 P 转发给所有终端. 终端用自己的私钥获取全局参数明文. 在参数上传阶段, 终端首先为本地参数添加满足差分隐私的噪声, 然后用中心服务器的公钥加密本地参数后发送给 P, P 收集所有更新后统一转发给服务器解密. 代理服务器 P 通过统一收集用户的加密参数后转发给中心服务器的方式隐藏了参数和终端之间的对应关系, 实现中心服务器和终端之间的匿名通信、保护本地隐私, 同时降低了中心服务器与用户之间多次通信的负担.

除了在安全多方计算协议上的探索以外, 噪声的添加方式和隐私的分析方式也受到关注. Hu 等人[108]借助安全聚合的方式保护参数在本地参数的隐私, 同时为梯度添加高斯噪声保护聚合参数在中心服务器处的隐私.为降低噪声带来的干扰, 该算法采用 zCDP 技术, 在考察梯度和本地模型的敏感度后为隐私损失提供更紧的边界, 能够同时支持强凸和非凸目标函数的优化. Gong 等人[109]在安全多方计算的基础上选择部分梯度上传,并且采用动态分配的方式压缩隐私预算, 提高模型的准确度. 此类方案借助安全多方计算保证了本地参数和数据的隐私, 通过本地加噪保证了中心参数和模型的隐私, 提高了模型的可用性. 但是上述方案中引入安全多方计算模块, 通信次数多、计算效率低的问题始终存在. 并且对于用户和中心服务器的可靠程度有了更多前提假设, 这些方案无法广泛应用于生产.

4.3.3 安全混洗

安全混洗(shuffling)为联邦学习的客户-服务器架构提供了隐私保护的新思路. 通过打乱终端提交的数据的顺序, 服务器仅能获取记录的乱序集合, 既达到了匿名的效果, 又保持了数据的准确性. 匿名是隐私保护的基础思想, 通过切断数据内容与数据提供者身份的关联性, 为数据提供隐私保护. Choudhury 等人[110]尝试了基于(k,k^m)匿名技术[111]的联邦学习隐私保护算法, 但是由于 k 匿名技术自身的不足, 该方法无法避免同质攻击和背景攻击, 而 k 匿名及其衍生技术也没有在联邦学习中得到广泛应用. 安全混洗的提出则为匿名思想提供

了全新的思路.

安全混洗的提出源于 2017 年, Google 公司率先设计了编码-混洗-分析框架(encode-shuffle-analyze, ESA)[112], 并在随后的工作中对框架细化[113]. ESA 框架如图 7 所示, 具体包含以下 3 个部分.

(1) 编码器(encoder): 编码器部署在终端, 本地通过分片、随机扰动或嵌套加密等方式处理原始记录后发送给混洗器. 分片能够避免记录因为某些特殊属性值的组合而被识别, 具体包括属性分片、发送分片、建模分片[114]等方式. 扰动和加密则负责保护本地记录的隐私.

(2) 混洗器(shuffler): 混洗器是独立于中心服务器的第三方服务器, 一般假设为半诚实服务器. 混洗器负责去除与身份相关的信息, 为收集的记录添加群 ID, 待数据量达到一定阈值 t 后, 按群 ID 分批混洗. 群 ID 使得类似的记录在同一个批次中被打乱、分析处理, 有效地提高了后续数据分析的准确性, 一般依据非敏感信息或哈希编码生成.

(3) 分析器(analyzer): 分析器部署在中心服务器上, 在 ESA 框架下, 一般假设为不可信服务器. 分析器对数据解密或无偏估计、存储、聚合并淘汰后分析和发布.

ESA 框架中, 由本地编码器和第三方混洗器联合为数据隐私提供保证, 抵御分析器和模型使用者发起的攻击. 在去除记录独特性、匿名用户身份的基础上, 该框架结合本地化差分隐私或者密码学工具后, 能够在保护隐私的同时保持数据的可用性. 上述方法在 ESA 系统中已经模块化, 每个模块都可以利用隐私保护机制进一步优化, 是一个可扩展的隐私保护方案.

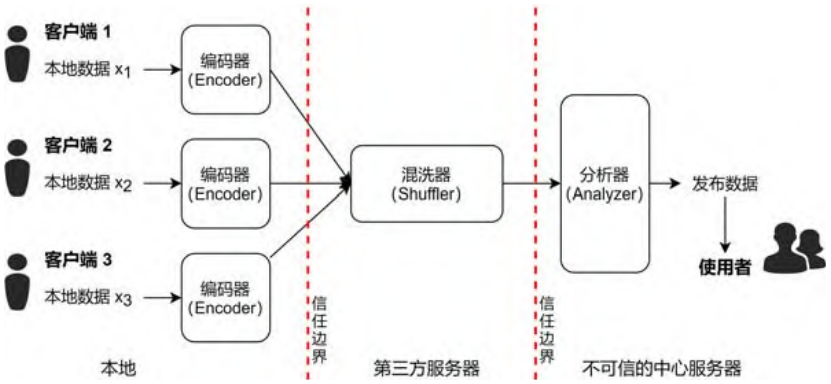

图 7 编码-混洗-分析(ESA)框架

安全混洗的具体实现方式很多. Kwon 等人[115]根据混合网络(mixnet)设计混洗器. 混洗器分别生成各自的产生密钥 k_i 并发送给用户, 用户利用密钥对自己的数据 x_i 嵌套加密得到 x_i' 后上传到混洗器 s_1, 该混洗器用自己的私钥 k_1 解密后将数据打乱, 发送给混洗器 s_2. 重复上述过程, 直到完成混洗. Bittau 等人[112]提出了一种在可信硬件平台 SGX 上的实现方式 PROCHLO. 由于可信硬件平台的内存有限, PROCHLO 引入了贮存缓冲区域, 每次随机抽取一桶数据并读取存储在缓冲区域的记录后一起打乱, 将溢出的记录暂存在该缓冲区域. 打乱的记录过滤加密后按批输出.

安全混洗的提出, 加强了隐私保护的效果. 尽管安全混洗本身只能提供匿名级别的隐私保护, 但是它与隐私技术的结合却能为本地和中心的数据提供强有力的保护. 目前, 基于安全混洗的隐私理论主要有两类: 一类基于本地化差分隐私和下采样放大理论, 一类基于中心化差分隐私和分割混合技术(split-mix). Erlingsson 等人[116]首次证明了安全混洗对本地化差分隐私的放大作用. 结合下采样隐私放大的结论[117], 作者通过分析一次元素顺序置换对本地隐私预算 ε_0 的放大效果, 证明了安全混洗能够使本地化差分隐私转化为中心化差分隐私. 其中, 中心的隐私预算 ε 降低 $\sqrt{n}$ 倍, 而中心的隐私预算 ε 相同时数据误差缩小 $\sqrt{n}$ 倍. Erlingsson 从理论上严格证明了经过安全混洗的本地化差分隐私能够为中心服务器提供更大程度的差分隐私保护, 中心无须再添加任何额外的噪声保护. 该工作以本地化差分隐私为基础, 为安全混洗的隐私放大作用提供了一般化的证

明. Ghazi 等人[118]首次基于分割混合技术实现了安全混洗中的中心化差分隐私保护. 其中, 分割混合技术[119]将终端的输入 x_i 分割为 m 个随机值 y_{ij}, 满足 $x_i=\sum y_{ij}(\mathrm{mod}\,q)$, 其中, $x_i, y_{ij}\in F_q$, F_q 是阶为 q 的加法群. 当分割消息的数量 m 超过特定阈值, 任意两个和相同的不同输入序列之间的方差均不超过 $2^{-\sigma}$. 借助此技术, 终端将分割碎片 y_{ij} 发送给混洗器打乱后, 几乎能够实现用户输入的完全匿名. Ghazi 等人证明了分割混合技术在安全混洗框架下能够实现差分隐私, 同时降低了分析器估计的误差上界和算法通信量, 并在后续工作中进一步降低误差期下界[120].

安全混洗协议可根据编码器对于输入数据的编码形式分为单消息协议和多消息协议两种, 其中, 单消息协议指编码器将一个输入编码为一条数据, 多消息协议指编码器将一个输入编码为多条数据并发送. Cheu 等人[121]提供了单消息协议的隐私放大证明, 在此场景下, 中心的隐私预算降低更多, 放大效果更加显著. 该方案分别设计了对布尔值和实数的保护方案, 用于计数估计和求和估计. 在计数估计中, 终端首先根据本地化差分隐私预算 ε_0 将原始数据的每一维随机扰动为 0 或 1 后上传至混洗器. 在求和估计中, 终端将原始数据按照无偏概率映射为多个 0, 1 向量后, 对每一维随机扰动, 分析器收集后进行无偏估计. 该算法经过混洗器, 中心隐私预算得到进一步放大. Balle 等人[122]同样设计了单消息发送协议, 为安全混洗的频数估计提供了另一种思路. 在最坏情况下, 攻击者能够识别所有上传数据中的真实值, 剩下的随机值分布被定义为“隐私毯子”. 当目标终端上传数据时, 算法保证真实信息能够被隐藏在隐私毯子之中. 作者进一步证明: 当选择合适的本地化差分隐私随机响应机制时, 针对频数估计的隐私毯子的隐私放大作用强于 Erlingsson 等人[116]的算法一般化证明.Balle 等人[123]则构建了多消息单论协议, 用于求和估计. 该方法以精度 p 将输入 x_i 映射为整数 $x_i'=\lfloor x_ip\rfloor+Ber(x_ip-\lfloor x_ip\rfloor)$后, 添加满足差分隐私的服从玻利亚分布(Pólya distribution, 可以由负二项分布生成)的噪声, 然后基于分割混合技术将 x_i' 分割为 m 个秘密后发送给 m 个混洗器. 最终, 分析器得到的估计值的均方误差(mean squared error, MSE)仅 $O(1/\varepsilon^2)$, 与用户数量 n 无关. 该方案借助多消息发送和分割混合技术提高了估计准确性, 但是付出了巨大的通信代价. Ghazi 等人[124]同样构建了多消息协议, 用于计数和求和估计. 终端分别生成与输入相关的扰动值和与输入无关的独立噪声, 该算法将相关扰动值和独立噪声联系起来, 确保扰动数据集分布的相似性, 首次在中心处实现了纯差分隐私保护. 作者在后续工作[125]中进一步基于“隐私毯子”的思想构建多消息协议用于频数估计, 允许终端向混洗器发送其本地数据和“隐私毯子噪声”来隐藏信息, 进一步降低了频数估计误差.

安全混洗目前仅在联邦学习的经验风险最小化模型上有简单应用[115,126]. Kwon 等人[115]在经验风险最小化模型(empirical risk minimization, ERM)的梯度下降中, 采用本地化差分隐私结合混洗的方案, 但要达到较高的准确性, 仍需大量迭代和较高的中心隐私预算, 效果并不理想. Liu 等人[126]则在安全混洗的基础上对高维梯度二次采样, 根据采样带来的隐私放大作用[127]进一步节约隐私预算. 并且该算法仅选择部分权重最高的梯度维度, 使得模型可用性大大提高. 但是二次采样和选择最高权重的方法是否具有一般性仍待进一步研究. 安全混洗依赖密码学工具, 计算开销大且对模型的准确性有一定的影响, 在联邦学习上的应用依然不够成熟.

4.4 隐私保护算法分析与对比

中心、本地、中心与本地同时保护这 3 类隐私保护策略既有联系又有区别, 而这 3 类隐私保护策略中蕴含的扰动、迁移、加密、安全计算与扰动结合、安全混洗等 5 类隐私保护方法又有各自的优缺点. 对比这 3 类隐私保护策略和 5 类保护方法的保护的对象、能够抵御的攻击、采用的技术、对模型训练的影响, 能够帮助我们为特定场景下的联邦学习设计合适的保护算法.

首先, 3 类隐私保护策略的保护对象和能够抵御的攻击不同.

(1) 中心保护策略的保护对象是中心服务器处的训练中间参数和发布的模型参数. 但此策略下本地参数缺乏保护, 无法抵御中心服务器发起的重建攻击. 基于扰动的中心保护策略中, 记录级别的差分隐私保护无法抵御成员推断攻击, 这是因为隐私定义在了记录而不是用户上.

(2) 本地保护策略的保护对象是终端的模型参数或即将公开的本地参数, 能够抵御联邦学习中特有的

内部隐私攻击. 该策略能够为本地参数提供严格的保护, 避免本地数据被中心服务器通过重建攻击获取. 但是对于全局参数和训练完成的模型则缺乏保障. 其中, 基于扰动的本地保护策略全局模型可用性得不到保障; 而基于加密的本地保护策略只针对数据的内部交换, 聚合的参数和发布的模型不受任何隐私保护. 因此, 内部攻击者能够根据全局参数发起推断攻击, 外部攻击者则能够根据模型参数发起重建攻击和推断攻击, 窃取训练集敏感信息.

(3) 上述两种策略都着重针对训练过程中的单个阶段提供保护, 没有作出整体性考虑. 中心与本地同时保护策略的目标则同时考虑了本地参数和全局参数, 能够同时抵御重建攻击和推断攻击, 但依赖特定的场景假设. 基于混合扰动的策略在中心服务器不可信时依然可能泄露隐私, 而其他扰动方案没有具体的实践应用;基于安全计算和安全混洗的扰动的方案则往往需要多种前提假设, 例如中心服务器半诚实、终端之间不共谋、终端半诚实等假设. 面对恶意攻击者主动篡改数据、影响训练过程的攻击方式更加脆弱, 且受制于密码学工具所带来的高昂计算代价和通信代价.

其次, 5 种隐私保护方法采用的技术和对模型训练的影响不同.

(1) 扰动方法基于中心化或本地化差分隐私技术实现, 在 3 类保护策略中都有应用. 扰动方法能够灵活地满足大部分隐私需求, 几乎不会带来额外的通信开销. 但是扰动机制仍然会降低模型准确度, 增加迭代收敛轮数.

(2) 迁移方法基于迁移学习思想和差分隐私技术, 主要应用于中心保护策略. 迁移方法的思想接近数据生成, 通过迁移学习将原始数据与训练数据分离达到隐私保护的效果. 迁移方法能够处理客户端模型不同的场景, 并且可以通过复杂模型向简单模型迁移, 提高中心模型的训练和预测效率. 但是这种方法需要在扰动待标注的训练集, 对模型的准确性影响较大. 而随着训练轮数的增加, 隐私开销也将急剧升高.

(3) 加密方法基于同态加密及秘密共享技术, 主要应用于本地保护策略、中心与本地同时保护策略中. 加密方法能够为训练过程中加密的对象提供严格的保护, 但是对于模型的后续发布无效. 由于同态加密技术仅支持整数运算、有限次乘法运算等限制, 基于同态加密的保护方法在集中式机器学习中计算准确度不高, 但在联邦学习中, 中心服务器仅对密文做聚合运算的场景下, 却可以达到较好的训练效果. 然而, 同态加密带来的高昂的计算、存储开销却限制了该方法的应用. 而秘密共享技术下的保护方法虽然降低了计算的代价, 通信代价却大大升高.

(4) 安全计算与扰动结合的方法基于安全多方计算技术, 需要中心化差分隐私的配合才能完成保护. 二者能够实现对模型从训练到发布的全周期保护, 但是扰动对于模型的准确性带来一定影响. 而基于密码学的安全计算不仅为模型带来极高的计算量和通信代价, 复杂的协议也缺乏通用性.

(5) 安全混洗方法基于加密以及差分隐私技术, 主要应用于中心与本地同时保护策略中. 安全混洗能够在保护本地数据隐私的同时, 在中心模型上达到较低的统计误差, 在可用性-隐私性上较为平衡. 但是该方案目前只能完成较为简单的数据统计, 在高维参数和模型训练的准确性上有待改善. 此外, 该方案侧重于加强保护中心服务器发布的模型, 当中心的隐私需求得到满足时, 由于隐私放大的作用, 本地隐私保护程度可能并不够, 需要额外的密码学工具保护. 因此, 为中心和本地同时挑选合适的隐私参数, 也是该方法需要面临的问题.

最后, 尽管目前的隐私防御策略和方法存在着种种不足, 但还是富有指导意义. 中心与本地同时保护的策略能够为联邦学习提供全周期的保护, 但是需要以中心及本地策略为基础; 而中心和本地在满足一定的信任条件或可用性要求时, 也有其适用场景. 中心扰动方法无法防御内部攻击, 但是迁移方法、安全计算与扰动结合方法、安全混洗方法都需要以中心扰动为基础设计, 中心扰动方法的研究价值依然很高. 本地扰动方法的可用性较低, 但是与安全混洗结合后, 其隐私性和可用性都得到大幅度提升. 而本地加密方法则为安全计算和安全混洗提供了一定的指导. 综上所述,扰动方法和加密方法为迁移、安全计算、安全混洗提供了具体的细节实现, 而迁移、安全计算、混洗则为扰动和加密方法的隐私性、可用性带来提升. 算法的对比详见表 4.

表 4 隐私保护算法对比

文献	保护策略	架构	保护提供者	保护对象	保护方法	技术	优点	缺点	抵御攻击
[53]	中心	客户-服务器	中心	中心梯度	扰动	中心化差分隐私	准确性较高、通信开销低、计算量较低、梯度可自适应加噪	需中心服务器可信	外部攻击
[54]									
[55]									
[56]									
[57]									
[59]			本地						
[60]									
[61]									
[62]									
[63]			本地和中心						外部攻击、部分内部推断攻击
[65]			中心	中心模型	知识迁移	迁移中心化差分隐私	可向简单模型迁移、客户端模型可异构	隐私开销与迭代轮数直接相关、需中心服务器可信	外部攻击
[66]									
[67]									
[69]									
[70]	本地	端对端	本地	本地模型	扰动	中心化差分隐私	通信开销低、扰动方案灵活、提供后续隐私保护、适用于所有架构	准确性低	内部重建、成员推断
[71]									
[75]									
[79]		客户-服务器		本地特征		本地化差分隐私			所有攻击
[85]									
[76]				本地梯度					
[82]									
[83]									
[87]									
[88]						本地化差分隐私随机投影			
[89]						本地化差分隐私 k 阶离散			
[91]					加密	同态加密	准确性高、隐私保护严格、支持客户端掉线	计算开销高、通信开销高、不保护发布模型	内部重建攻击
[93]						秘密共享			
[94]									
[95]						同态加密秘密共享			
[96]									
[97]	中心和本地		中心和本地	部分本地梯度，部分中心梯度	扰动	中心化差分隐私本地化差分隐私	全阶段保护、准确性较高、通信开销低、计算量较低	通用性不足	外部攻击、部分内部攻击
[98]				本地和中心梯度		中心化差分隐私函数机制			所有攻击
[99]				本地数据集大小本地噪声大小中心模型	安全计算与扰动	安全多方计算中心化差分隐私	全阶段保护、准确性较高、隐私保护严格	计算开销高、通信开销高、协议复杂脆弱、需终端不共谋等假设	外部攻击
[100]									
[101]				本地梯度中心梯度					所有攻击
[104]									
[106]									
[107]									
[109]									
[108]									
[110]					匿名	(k,k^m)匿名	方案简单	存在隐私泄露隐患	
[112]					安全混洗	安全混洗分割混合技术本地化差分隐私	全阶段保护、准确性较高、隐私保护严格、隐私开销低于本地扰动策略	计算开销高、在高维参数的保护上不成熟	
[113]									
[116]									
[118]									
[120]									
[121]									
[122]									
[123]									
[124]									
[125]									
[126]						安全混洗二次采样本地化差分隐私			

5 未来展望

不同于传统的集中式机器学习, 联邦学习由于自身架构和训练方式的独特性, 面临着更多样的隐私攻击手段和更迫切隐私保护需求. 现有的联邦学习隐私保护算法在技术、平衡性、隐私保护成本和实际应用中还存在诸多不足之处. 明确这些问题和挑战, 才能展望联邦学习隐私保护未来发展的机遇和方向.

5.1 构建隐私量化体系,设计有针对性的隐私定义和保护技术

尽管对联邦学习隐私保护方案的探索已经不少, 但是目前, 方案中所应用的隐私保护技术却无法满足联邦学习独特的隐私保护需求. 由于针对联邦学习的隐私保护程度不明确、目标不清晰、方式单一, 导致方案提供的保护或者过于严格进而影响模型的性能, 或者程度弱且保护对象模糊, 或者保护目标单一难以达到隐私性与可用性平衡.

目前的隐私保护技术还存在以下问题.

(1) 现有的隐私技术保护程度不明确. 利用差分隐私保护隐私的方案虽然能够明确隐私开销, 但是隐私开销与用户的隐私需求、抵御攻击的能力大小之间缺乏定量关系. 例如, 用户级别的差分隐私虽然能够保护一个用户的多条记录, 但是对其防御能力缺乏量化验证, 用户级别和记录级别保护技术的保护能力也就无法进一步分析.

(2) 现有的隐私技术防御目标不清晰. 现有的隐私保护技术虽然都以保护数据为主要任务, 但是与联邦学习隐私防御的目标不能完全匹配: 利用差分隐私技术提供隐私保护的方式本质上是保证训练集的成员记录不被识别, 因此差分隐私理论仅能抵御成员推断攻击, 对于重建攻击、属性推断攻击的抵御能力尚缺乏明确的证明; 利用密码学工具提供隐私保护的方式无法抵御大部分内部隐私攻击, 必须借助差分隐私技术共同完成保护, 模型发布后同样面临上述问题.

(3) 现有的联邦学习隐私保护方式单一、缺乏针对性. 联邦学习在本地参数上传、全局参数下发、终端之间参数交换等不同阶段面临的风险不同, 但由于缺乏明确的隐私程度度量和隐私防御目标设定, 现有技术无法根据各个阶段和数据情况设计保护方案: 中心化差分隐私和本地加密等技术对本地参数上传几乎没有保护; 而本地化差分隐私能够保护本地参数上传, 但对于后续的全局参数保护程度过高, 影响模型的可用性. 即使是二者相结合的中心与本地同时保护策略, 也往往只能为训练全周期提供同样标准和程度的隐私保护, 导致联邦学习在不同阶段实际抵御攻击的能力不同.

要完善针对联邦学习隐私技术, 可以考虑以下方向.

(1) 构建隐私量化体系. 从联邦学习隐私攻击的角度出发, 量化隐私保护技术的抵御攻击的能力; 从数据的角度出发, 量化联邦学习用户对不同内容的记录和数据集的隐私需求, 从而明确联邦学习中不同的数据层次、防御层次的隐私需求, 完善隐私量化体系.

(2) 设计有针对性的隐私定义. 可以根据联邦学习各个阶段参数泄露的可能性、各个迭代轮数间隐私泄露风险不同, 设计细分隐私定义, 为不同的训练阶段、迭代轮数分配合理的隐私开销; 根据联邦学习面临的多种隐私攻击, 量化隐私风险, 为不同类型的攻击设计有针对性的隐私定义.

(3) 探索有针对性的隐私保护技术. 有针对性的隐私定义为细粒度分阶段制定隐私方案带来可能, 而隐私量化体系则明确隐私保护技术与攻击以及用户需求之间的数量关系. 在此基础上, 可以探索更有针对性的保护手段, 从而结合联邦学习模型和数据的具体情况, 有重点地保护数据集中高隐私需求的部分. 例如, 基于信息论的隐私保护[128]技术能够通过度量输出结果与非敏感信息的交叉熵, 使训练模型与敏感属性的关联性尽可能降低, 同时不影响非敏感数据对模型的贡献. 结合隐私量化和定义探索针对联邦学习具体场景的隐私保护技术, 能够在严格保护隐私的同时保持模型可用性.

5.2 研究隐私性、鲁棒性、公平性合一的隐私保护机制

尽管学术界正在不断探索联邦学习的隐私保护方案, 但是常见的方案为了保护隐私, 模型的鲁棒性(robustness)和公平性(fairness)往往会受到影响. 此处的鲁棒性是指模型容忍抵御恶意攻击稳定运行的能力,

公平性则指模型平等对待具有相同特征的个体或群体的能力. 如何缓解联邦学习隐私保护方案对鲁棒性和公平性的负面影响, 寻求能够平衡三者关系的统一的解决方案, 是联邦学习隐私保护需要面对的挑战.

隐私保护带来的影响如下.

(1) 目前的隐私保护方案很少考虑对模型鲁棒性的影响. 然而联邦学习场景下, 由于参与训练的角色增加, 潜在攻击者能够篡改数据并获取中间参数, 模型被恶意攻击的风险提高. 以加密方案为例, 由于隐私协议的存在, 中心服务器只能获取参数的聚合结果而无法获取单个终端的参数. 因此, 面临恶意攻击时, 中心服务器无法识别恶意参与者, 鲁棒性大大降低; 反之,鲁棒性高的模型在面对隐私攻击时也更加脆弱[129]. 如何在隐私保护的同时不损害模型对抗恶意攻击的鲁棒性, 是联邦学习隐私保护需要考虑的难题.

(2) 隐私性与公平性无法兼容. 由于数据生成和收集时的偏差, 联邦学习的模型输出结果很容易倾向于拥有大量样本的类别、低延迟且算力更优的终端. 尽管 Lyu 等人[130]考虑了不同设备对于模型的贡献程度, 但依然存在更多因素影响公平性. 解决联邦学习的公平性, 需要训练模型时获取人口统计信息和不同终端的设备信息, 从而为少量样本、算力不足的终端分配合理的训练方式或权重. 然而在隐私保护中, 人口统计信息往往属于敏感信息, 如性别、年龄、种族等. 数据持有者为尽量避免暴露敏感信息给第三方, 常选择非敏感属性训练或掩盖敏感属性, 致使其他参与方无法获取人口统计信息. 而终端算力带来的结果偏差则由于实际训练情况复杂, 难以得到有效的校正. 因此, 隐私保护模型难以简单地依据人口统计或设备等信息保证公平性.

为平衡隐私性、鲁棒性和公平性之间的矛盾, 可以从以下几个角度考虑:

(1) 探究同时满足隐私性和鲁棒性的解决方案. 由于引入了噪声, 通过差分隐私保护训练的模型的稳定性得到一定的提高. 例如, 针对数据投毒等恶意攻击, 篡改少量样本为经过差分隐私保护的模型带来的影响可能小于一般模型. 因此, 量化差分隐私对于投毒攻击的保护程度, 是一种协调隐私与鲁棒性的平衡性的可能方式.

(2) 探索不依赖敏感信息和设备信息实现公平性的方式. 在联邦学习隐私保护的过程中, 针对不同级别、不同数量的样本, 生成与敏感信息无关的小样本的嵌入式表示, 使得数量不占优势的类别得到模型充分的重视, 又不泄露隐私; 根据终端多次发送的参数生成终端的嵌入式表示, 用以对比区分终端的设备算力, 充分考虑不同设备间的公平性.

(3) 寻求平衡隐私性、鲁棒性和公平性合一的模型. 关注数据拥有者不同的需求, 制定个性化、自适应的方案, 从而同时满足隐私性、鲁棒性和公平性的要求.

5.3 实现低成本、轻量级的联邦学习隐私保护策略

联邦学习中数据留在本地, 负面作用是带来了大量参数交换. 在此基础上, 隐私保护成本也被抬高. 因此, 实现低成本、轻量级的联邦学习隐私保护策略, 是未来发展的必然阶段.

联邦学习隐私防御成本包括:

(1) 更高的通信代价和计算开销. 基于安全多方计算或加密的方案以密文计算为基础, 需要更多的数据量传输、安全证明, 也带来了更加高的计算量, 目前无法扩展到大规模复杂模型的训练上来. 基于本地化差分隐私的保护方案虽然不会带来额外的通信代价, 但是目前的本地化差分隐私技术需要千万级样本量才能实现中心服务器的无偏估计, 该方案隐含的计算开销非常高.

(2) 模型准确性降低. 基于差分隐私的方案需要在训练过程中添加大量噪声, 而基于加密或安全多方计算的方案在训练复杂网络时需要对激活函数线性近似. 这些都对模型的准确性造成一定影响. 尽管差分隐私能够部分增强模型的泛化能力, 但是具体增强程度缺乏量化衡量指标.

(3) 协议复杂脆弱. 以安全多方计算为基础的隐私保护需要多个参与者共同完成, 当参与者中出现恶意攻击者、突然掉线等设定外的状况时, 协议极易被破坏导致训练失败. 为了避免这些状况的出现, 有些方案设计了对应措施, 但是这些措施也进一步加剧了协议的复杂性, 提高了通信代价. 差分隐

私方案虽然能够支持参与者掉线及异步训练, 并能容忍少量恶意数据, 但是在训练准确性上低于同步训练的方案.

为实现低成本、轻量级的隐私保护策略, 可以从以下几个方面展开.

(1) 利用隐私放大技术降低隐私保护的成本: 通过安全混洗, 将联邦学习的本地隐私转化为中心隐私预算, 相比于分别在中心和本地添加保护, 隐私预算大大降低; 量化训练过程中, 终端随机选择样本和服务器随机选择终端的内在隐私性. 利用训练过程中已经存在的随机性提供保护, 避免添加过多、过于严格的额外保护.

(2) 探索轻量级隐私保护方案: 除了本文中阐述的隐私保护技术以旬, 还可以采用更加轻量级的技术提供保护, 降低模型训练过程的复杂性、降低隐私保护带来的额外计算量和通信代价. 例如采用秘密共享技术交换参数, 量化泛化、降维、利用内在随机性等方式为模型带来的保护程度等.

(3) 设计自适应的隐私联邦训练方案: 通过优化联邦学习的训练过程, 降低参数的交换量、模型的聚合次数, 在提升训练效率的同时, 降低隐私开销.

5.4 探索面向复杂场景的异质联邦学习隐私保护方案

目前主要的研究都针对横向联邦学习, 尤其是客户-服务器架构. 然而横向联邦学习中, 每个终端的数据分布、特征属性都相同, 数据规模近似, 数据量充足, 只是一种理想的假设. 在实践中, 联邦学习往往面临数据上和系统上的异质: 数据异构、不同分布、不同特征. 模型结构和联邦架构多样、实际应用场景丰富. 因此, 不局限于理想假设, 探索异质联邦学习的隐私保护方案, 是联邦学习真正落地的必经之路.

联邦学习在系统上的隐私保护研究不足.

(1) 不同的模型结构: 目前的方案都基于一个假设, 即所有终端训练的模型结构相同, 且与中心服务器相同. 针对各个终端模型类型不同、结构不同的训练场景, 还没有更多研究.

(2) 不同的联邦架构: 尽管针对端对端的隐私保护方案也有研究涉猎, 但是大部分方案都采用直接由一个终端发布完整模型后交付的策略, 协同训练的保护方案缺乏深入研究.

联邦学习在数据分布上的隐私保护困境表现在:

(1) 纵向联邦学习: 目前已有少量研究借助同态加密、差分隐私等技术为纵向联邦学习提供隐私保护方案, 其保护思路与保护横向联邦学习类似. 然而纵向场景下, 各方掌握的特征各异, 每一次梯度计算都需要各方多次交换中间参数, 其计算量和通信量本身已高于横向联邦学习. 为满足隐私需求, 保护中间参数又进一步恶化了这一情况. 例如, 加密的保护方法[131]带来了巨大的数据量, 进一步抬高通信代价; 扰动的保护方法[73,132]则需根据具体的中间参数的形式计算差分隐私参数, 其计算和证明过程繁复且不具通用性.

(2) 迁移联邦学习: 类似地, 迁移联邦学习由于损失函数更复杂、协同训练时中间参数交换更多, 面临的效率问题也更加严重[92,133]. 基于同态加密的保护方案通信量过高, 基于秘密共享的保护方案通信次数激增. 为便于中间参数交换, 可能还需要近似展开目标函数, 影响模型的收敛效率和准确性.

上述两种联邦学习在训练前都需要完成实体对齐等预处理, 同样缺乏高效的隐私保护手段.

此外, 联邦学习在不同应用场景上的隐私保护研究也存在不足: 目前的研究都基于大量终端协同训练. 已有研究证明: 终端数量越高, 隐私保护下的可用性越高. 终端数量较少时, 现有技术将对模型准确性造成严重损伤. 终端较少时的低可用性, 限制了面向企业(to B)的联邦学习隐私保护发展.

未来可以在以下场景中探索隐私保护方案.

(1) 非独立同分布的数据上的隐私保护. 在隐私保护的前提下, 处理样本和数据量偏斜、消解终端内及终端间数据的相关性, 同时避免隐私保护加剧模型不收敛等现象.

(2) 纵向联邦学习的隐私保护. 消除不同损失函数的影响, 在理论上构建通用的中间参数扰动方案, 同时尝试选择发送部分参数、压缩参数数据量, 降低通信代价; 考虑到一方掌握全部标签的情况, 研究者需要为知识更少、攻击能力更弱的参与方分配更多的隐私预算, 设计合理的隐私分配机制; 有

效结合加密和扰动方法, 达到全周期保护效果的同时, 降低扰动方案隐私参数计算对中间参数的依赖性.

(3) 迁移联邦学习的隐私保护. 探索可扩展到大规模、多参与方的加密方案; 探索更加丰富的迁移目标函数, 降低训练和中间参数复杂性, 避免隐私保护加剧模型计算的复杂性.

(4) 隐私预算的个性化定制. 综合考虑每个终端不同的隐私保护需求、诚实程度、信任第三方的意愿以及设备的计算和通信能力, 制定个性化、多层次的隐私保护方案, 实现隐私按需分配, 自适应地处理更加复杂的实际应用场景.

5.5 解决高维中间参数的隐私隐患

由于样本维度高、模型结构复杂, 联邦学习过程中交换的参数往往具有极高的维度, 不仅带来了巨大的通信代价, 而且面临着严重的隐私隐患. 如何解决高维中间参数的泄露风险、提高隐私保护的可靠程度, 是当前联邦学习隐私保护研究聚焦的关键问题之一.

由于模型的结构过于复杂、梯度维数过高, 仅仅依靠有限的样本量梯度无法很好地泛化. 高维参数能够携带大量样本信息, 一旦被窃取, 将泄露原始数据的信息. 然而, 高维参数的隐私保护仍然存在诸多问题: 由于高维参数携带的信息丰富, 基于中心化差分隐私的扰动并不能抵御重建攻击; 而更加严格的基于本地化差分隐私的方案需要根据维度的数量拆分隐私开销, 致使每一维分得的隐私预算低, 引入噪声量大, 严重影响模型训练的收敛和最终的准确性; 基于加密的方案中, 维度的增加带来更多计算开销, 难以在大规模训练中应用. 此外, 高维参数也会为模型带来极高的通信代价, 影响模型性能.尽管有研究通过降维保护的方案降低隐私和通信开销. 但是降维后的参数无法被接收方完全还原, 对模型的准确性造成一定影响. 因此, 高维中间参数给隐私保护、模型可用性和通信代价带来的负面影响亟需解决.

解决高维度中间参数的隐私隐患, 可以从以下 3 个方面展开.

(1) 探索本地化差分隐私对于高维参数的保护方式, 结合轻量级安全计算技术, 降低本地化差分隐私引入的噪声量.

(2) 探索高维参数无损压缩和恢复的机制, 在压缩的参数上扰动, 收紧隐私预算.

(3) 寻求更多训练方式. 目前, 主要的保护方式都基于模型平均和梯度平均两种联邦学习算法, 不可避免地面临高维中间参数的问题. 可以考虑寻找更多方案, 例如引入迁移学习、模型压缩等技术, 达到联合学习、分享模型的目的.

6 总 结

联邦学习的提出和兴起, 为解决数据孤岛、数据隐私带来巨大转机. 但是随着参与训练的角色增多、能力增强, 联邦学习也由此面临着与集中式机器学习不同的隐私泄露风险. 为联邦学习提供隐私保护方法, 是联邦学习进一步发展的必经之路, 也是联邦学习应用落地的基石.

本文针对联邦学习的隐私泄露风险和保护的最新研究做出了充分调研和深入分析, 介绍了联邦学习的架构和类型, 分析了隐私风险的来源, 总结了主要攻击策略. 按照隐私保护对象归纳保护策略, 总结各个策略的不足之处, 为联邦学习的隐私保护梳理脉络. 最后, 根据已有研究讨论了未来的研究发展. 虽然目前针对联邦学习隐私保护的研究已有不少, 但是适用场景单一、背景假设过强、抵御攻击的能力有限, 仍有许多问题亟需解决, 远未达到成熟的水平. 联邦学习隐私保护仍然等待着更多广泛、深入的研究.

References:

[1] McMahan HB, Moore E, Ramage D, Hampson S, Arcas BA. Communication-efficient learning of deep networks from decentralized data. In: Proc. of the 20th Int'l Conf. on Artificial Intelligence and Statistics. 2017. 1273−1282.

[2] Li T, Sahu AK, Talwalkar A, Smith V. Federated learning: Challenges, methods, and future directions. IEEE Signal Processing Magazine, 2020, 37(3): 50−60.

[3] Zhu L, Liu ZJ, Han S. Deep leakage from gradients. In: Advances in Neural Information Processing Systems. MIT Press, 2019. 14774−14784.

[4] Song C, Ristenpart T, Shmatikov V. Machine learning models that remember too much. In: Proc. of the 2017 ACM SIGSAC Conf. on Computer and Communications Security. ACM, 2017. 587−601.

[5] Liu RX, Chen H, Guo RY, Zhao D, Liang WJ, Li CP. Survey on privacy attacks and defenses in machine learning. Ruan Jian Xue Bao/Journal of Software, 2020, 31(3): 866−892 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5904.htm [doi: 10.13328/j.cnki.jos.005904]

[6] Lyu L, Yu H, Yang Q. Threats to federated learning: A survey. arXiv preprint arXiv:2003.02133, 2020.

[7] Wang JZ, Kong LW, Huang ZC, Chen LJ, Liu Y, Lu CX, Xiao J. Research advances on privacy protection of federated learning. Journal of Big Data, 2021, 7(3): 130−149 (in Chinese with English abstract). http://kns.cnki.net/kcms/detail/10.1321.G2.20210112.1017.002.html

[8] Li Q, Wen Z, He B. A survey on federated learning systems: Vision, hype and reality for data privacy and protection. arXiv preprint arXiv:1907.09693, 2019.

[9] Vepakomma P, Swedish T, Raskar R, Gupta O, Dubey A. No peek: A survey of private distributed deep learning. arXiv preprint arXiv:01812.03288, 2018.

[10] Zhang D, Chen X, Wang D, Shi J. A survey on collaborative deep learning and privacy-preserving. In: Proc. of the 3rd IEEE Int'l Conf. on Data Science in Cyberspace (DSC). IEEE, 2018. 652−658.

[11] Yang Q, Liu Y, Chen T, Tong Y. Federated machine learning: Concept and applications. ACM Trans. on Intelligent Systems and Technology (TIST), 2019, 10(2): 1−19.

[12] Kairouz P, McMahan HB, Avent B, Bellet A, Bennis M, Bhagoji AN, Zhao S. Advances and open problems in federated learning. arXiv preprint arXiv:1912.04977, 2019.

[13] McMahan HB, Moore E, Ramage D, Arcas BA. Federated learning of deep networks using model averaging. arXiv preprint arXiv:1602.05629, 2016.

[14] Hegedűs I, Danner G, Jelasity M. Gossip learning as a decentralized alternative to federated learning. In: Proc. of the IFIP Int'l Conf. on Distributed Applications and Interoperable Systems. Cham: Springer, 2019. 74−90.

[15] Cheng K, Fan T, Jin Y, Liu Y, Chen T, Papadopoulos D, Yang Q. Secureboost: A lossless federated learning framework. IEEE Intelligent Systems, 2021, 36(6): 87−98.

[16] Hu Y, Niu D, Yang J, Zhou S. FDML: A collaborative machine learning framework for distributed features. In: Proc. of the 25th ACM SIGKDD Int'l Conf. on Knowledge Discovery & Data Mining. ACM, 2019. 2232−2240.

[17] Liu Y, Kang Y, Xing C, Chen T, Yang Q. A secure federated transfer learning framework. IEEE Intelligent Systems, 2020, 35(4): 70−82.

[18] Tramèr F, Zhang F, Juels A, Reiter MK, Ristenpart T. Stealing machine learning models via prediction apis. In: Proc. of the 25th USENIX Security Symp. (USENIX Security 2016). USENIX Association, 2016. 601−618.

[19] Song C, Ristenpart T, Shmatikov V. Machine learning models that remember too much. In: Proc. of the 2017 ACM SIGSAC Conf. on Computer and Communications Security. ACM, 2017. 587−601.

[20] Fredrikson M, Lantz E, Jha S, Lin S, Page D, Ristenpart T. Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing. In: Proc. of the 23rd USENIX Security Symp. (USENIX Security 2014). USENIX Association, 2014. 17−32.

[21] 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. ACM, 2015. 1322−1333.

[22] Shokri R, Stronati M, Song C, Shmatikov V. Membership inference attacks against machine learning models. In: Proc. of the 2017 IEEE Symp. on Security and Privacy (SP). IEEE, 2017. 3−18.

[23] 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. ACM, 2017. 603−618.

[24] Wang Z, Song M, Zhang Z, Song Y, Wang Q, Qi H. Beyond inferring class representatives: User-level privacy leakage from federated learning. In: Proc. of the IEEE INFOCOM 2019—IEEE Conf. on Computer Communications. IEEE, 2019. 2512−2520.

[25] Song M, Wang Z, Zhang Z, Song Y, Wang Q, Ren J, Qi H. Analyzing user-level privacy attack against federated learning. IEEE Journal on Selected Areas in Communications, 2020, 38(10): 2430−2444.

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

[27] Geiping J, Bauermeister H, Dröge H, Moeller M. Inverting gradients—How easy is it to break privacy in federated learning? arXiv preprint arXiv:2003.14053, 2020.

[28] He Z, Zhang T, Lee RB. Model inversion attacks against collaborative inference. In: Proc. of the 35th Annual Computer Security Applications Conf. IEEE, 2019. 148−162.

[29] Pan X, Zhang M, Ji S, Yang M. Privacy risks of general-purpose language models. In: Proc. of the 2020 IEEE Symp. on Security and Privacy (SP). IEEE, 2020. 1314−1331.

[30] Melis L, Song C, De Cristofaro E, Shmatikov V. Exploiting unintended feature leakage in collaborative learning. In: Proc. of the 2019 IEEE Symp. on Security and Privacy (SP). IEEE, 2019. 691−706.

[31] 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 (SP). IEEE, 2019. 739−753.

[32] Luo X, Wu Y, Xiao X, Ooi BC. Feature inference attack on model predictions in vertical federated learning. In: Proc. of the 2021 IEEE 37th Int'l Conf. on Data Engineering (ICDE). IEEE, 2021. 181−192.

[33] Orekondy T, Oh SJ, Schiele B, Fritz M. Understanding and controlling user linkability in decentralized learning. arXiv preprint arXiv:1805.05838, 2018.

[34] Jagielski M, Ullman J, Oprea A. Auditing differentially private machine learning: How private is private SGD? arXiv preprint arXiv:2006.07709, 2020.

[35] Dwork C, McSherry F, Nissim K, Smith A. Calibrating noise to sensitivity in private data analysis. In: Proc. of the Theory of Cryptography Conf. Berlin, Heidelberg: Springer, 2006. 265−284.

[36] McSherry F, Talwar K. Mechanism design via differential privacy. In: Proc. of the 48th Annual IEEE Symp. on Foundations of Computer Science (FOCS 2007). IEEE, 2007. 94−103.

[37] Nikolov A, Talwar K, Zhang L. The geometry of differential privacy: The sparse and approximate cases. In: Proc. of the 45th Annual ACM Symp. on Theory of Computing. ACM, 2013. 351−360.

[38] Ye QQ, Meng XF, Zhu MJ, Huo Z. Survey on local differential privacy. Ruan Jian Xue Bao/Journal of Software, 2018, 29(7): 1981−2005 (in Chinese with English abstract). http.//www.jos.org.cn/1000-9825/5364.htm [doi: 10.13328/j.cnki.jos.005364]

[39] Warner SL. Randomized response: A survey technique for eliminating evasive answer bias. Journal of the American Statistical Association, 1965, 60(309): 63−69.

[40] Dwork C, Roth A. The algorithmic foundations of differential privacy. Foundations and Trends in Theoretical Computer Science, 2014, 9(3-4): 211−407.

[41] Chaudhuri K, Monteleoni C. Privacy-preserving logistic regression. In: Advances in Neural Information Processing Systems. MIT Press, 2008. 289−296.

[42] Chaudhuri K, Monteleoni C, Sarwate AD. Differentially private empirical risk minimization. Journal of Machine Learning Research, 2011, 12(3): 1069−1109.

[43] Abadi M, Chu A, Goodfellow I, McMahan HB, Mironov I, Talwar K, Zhang L. Deep learning with differential privacy. In: Proc. of the 2016 ACM SIGSAC Conf. on Computer and Communications Security. ACM, 2016. 308−318.

[44] Yu D, Zhang H, Chen W, Liu TY, Yin J. Gradient perturbation is underrated for differentially private convex optimization. arXiv preprint arXiv:1911.11363, 2019.

[45] Wu N, Farokhi F, Smith D, Kaafar MA. The value of collaboration in convex machine learning with differential privacy. In: Proc. of the 2020 IEEE Symp. on Security and Privacy (SP). IEEE, 2020. 304−317.

[46] Rivest RL, Adleman L, Dertouzos ML. On data banks and privacy homomorphisms. Foundations of Secure Computation, 1978, 4(11): 169−180.

[47] Menezes AJ, Van Oorschot PC, Vanstone SA. Handbook of Applied Cryptography. CRC Press, 2018.

[48] Gentry C, Halevi S. Implementing gentry's fully-homomorphic encryption scheme. In: Proc. of the Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Berlin, Heidelberg: Springer, 2011. 129−148.

[49] Shamir A. How to share a secret. Communications of the ACM, 1979, 22(11): 612−613.

[50] Blakley GR. Safeguarding cryptographic keys. In: Proc. of the Int'l Workshop on Managing Requirements Knowledge. IEEE Computer Society, 1979. 313−318.

[51] Asmuth C, Bloom J. A modular approach to key safeguarding. IEEE Trans. on Information Theory, 1983, 29(2): 208−210.

[52] Paillier P. Public-key cryptosystems based on composite degree residuosity classes. In: Proc. of the Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Berlin, Heidelberg: Springer, 1999. 223−238.

[53] Geyer RC, Klein T, Nabi M. Differentially private federated learning: A client level perspective. arXiv preprint arXiv:1712.07557, 2017.

[54] McMahan HB, Ramage D, Talwar K, Zhang L. Learning differentially private recurrent language models. In: Proc. of the Int'l Conf. on Learning Representations. 2018. 1−14.

[55] McMahan HB, Andrew G, Erlingsson U, Chien S, Mironov I, Papernot N, Kairouz P. A general approach to adding differential privacy to iterative training procedures. arXiv preprint arXiv:1812.06210, 2018.

[56] Liu Y, Suresh AT, Yu F, Kumar S, Riley M. Learning discrete distributions: User vs item-level privacy. In: Advances in Neural Information Processing Systems. MIT Press, 2020. 20965−20976.

[57] Liang Z, Wang B, Gu Q, Osher S, Yao Y. Exploring private federated learning with laplacian smoothing. arXiv preprint arXiv:2005.00218, 2020.

[58] Osher S, Wang B, Yin P, Luo X, Barekat F, Pham M, Lin A. Laplacian smoothing gradient descent. arXiv preprint arXiv:1806.06317, 2018.

[59] Shokri R, Shmatikov V. Privacy-preserving deep learning. In: Proc. of the 22nd ACM SIGSAC Conf. on Computer and Communications Security. ACM, 2015. 1310−1321.

[60] Wu X, Li F, Kumar A, Chaudhuri K, Jha S, Naughton J. Bolt-on differential privacy for scalable stochastic gradient descent-based analytics. In: Proc. of the 2017 ACM Int'l Conf. on Management of Data. ACM, 2017. 1307−1322.

[61] Hu R, Guo Y, Li H, Pei Q, Gong Y. Personalized federated learning with differential privacy. IEEE Internet of Things Journal, 2020, 7(10): 9530−9539.

[62] Huang Z, Hu R, Guo Y, Chan-Tin E, Gong Y. DP-ADMM: ADMM-based distributed learning with differential privacy. IEEE Trans. on Information Forensics and Security, 2019, 15: 1002−1012.

[63] Wei K, Li J, Ding M, Ma C, Yang HH, Farokhi F, Poor HV. Federated learning with differential privacy: Algorithms and performance analysis. IEEE Trans. on Information Forensics and Security, 2020, 15: 3454−3469.

[64] Breiman L. Bagging predictors. Machine Learning, 1996, 24(2): 123−140.

[65] Hamm J, Cao Y, Belkin M. Learning privately from multiparty data. In: Proc. of the Int'l Conf. on Machine Learning. ACM, 2016. 555−563.

[66] Papernot N, Abadi M, Erlingsson U, Goodfellow I, Talwar K. Semi-supervised knowledge transfer for deep learning from private training data. arXiv preprint arXiv:1610.05755, 2016.

[67] Papernot N, Song S, Mironov I, Raghunathan A, Talwar K, Erlingsson Ú. Scalable private learning with PATE. arXiv preprint arXiv:1802.08908, 2018.

[68] Mironov I. Rényi differential privacy. In: Proc. of the 2017 IEEE 30th Computer Security Foundations Symp. (CSF). IEEE, 2017. 263−275.

[69] Sun L, Zhou Y, Yu PS, Xiong C. Differentially private deep learning with smooth sensitivity. arXiv preprint arXiv:2003.00505, 2020.

[70] Zhao L, Ni L, Hu S, Chen Y, Zhou P, Xiao F, Wu L. Inprivate digging: Enabling tree-based distributed data mining with differential privacy. In: Proc. of the IEEE INFOCOM 2018—IEEE Conf. on Computer Communications. IEEE, 2018. 2087−2095.

[71] Ding J, Wang J, Liang G, Bi J, Pan M. Towards plausible differentially private ADMM based distributed machine learning. In: Proc. of the 29th ACM Int'l Conf. on Information & Knowledge Management. ACM, 2020. 285−294.

[72] Bun M, Steinke T. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In: Proc. of the Theory of Cryptography Conf. Berlin, Heidelberg: Springer, 2016. 635−658.

[73] Wang C, Liang J, Huang M, Bai B, Bai K, Li, H. Hybrid differentially private federated learning on vertically partitioned data. arXiv preprint arXiv:2009.02763, 2020.

[74] Kearns M, Pai M, Roth A, Ullman J. Mechanism design in large games: Incentives and privacy. In: Proc. of the 5th Conf. on Innovations in Theoretical Computer Science. Elsevier, 2014. 403−410.

[75] Lu Y, Huang X, Dai Y, Maharjan S, Zhang Y. Differentially private asynchronous federated learning for mobile edge computing in urban informatics. IEEE Trans. on Industrial Informatics, 2019, 16(3): 2134−2143.

[76] Wang N, Xiao X, Yang Y, Zhao J, Hui SC, Shin H, Yu G. Collecting and analyzing multidimensional data with local differential privacy. In: Proc. of the 2019 IEEE 35th Int'l Conf. on Data Engineering (ICDE). IEEE, 2019. 638−649.

[77] Geng Q, Kairouz P, Oh S, Viswanath P. The staircase mechanism in differential privacy. IEEE Journal of Selected Topics in Signal Processing, 2015, 9(7): 1176−1184.

[78] Duchi JC, Jordan MI, Wainwright MJ. Minimax optimal procedures for locally private estimation. Journal of the American Statistical Association, 2018, 113(521): 182−201.

[79] Arachchige PCM, Bertok P, Khalil I, Liu D, Camtepe S, Atiquzzaman M. Local differential privacy for deep learning. IEEE Internet of Things Journal, 2020, 7(7): 5827−5842.

[80] Erlingsson Ú, Pihur V, Korolova A. Rappor: Randomized aggregatable privacy-preserving ordinal response. In: Proc. of the 2014 ACM SIGSAC Conf. on Computer and Communications Security. ACM, 2014. 1054−1067.

[81] Wang T, Blocki J, Li N, Jha S. Locally differentially private protocols for frequency estimation. In: Proc. of the 26th USENIX Security Symp. (USENIX Security 2017). USENIX Association, 2017. 729−745.

[82] Zhao Y, Zhao J, Yang M, Wang T, Wang N, Lyu L, Lam KY. Local differential privacy based federated learning for Internet of Things. IEEE Internet of Things Journal, 2020, 8(11): 8836−8853.

[83] Truex S, Liu L, Chow KH, Gursoy ME, Wei W. LDP-Fed: Federated learning with local differential privacy. In: Proc. of the 3rd ACM Int'l Workshop on Edge Systems, Analytics and Networking. ACM, 2020. 61−66.

[84] Gursoy ME, Tamersoy A, Truex S, Wei W, Liu L. Secure and utility-aware data collection with condensed local differential privacy. IEEE Trans. on Dependable and Secure Computing, 2019, 18(5): 2365−2378.

[85] Wang Y, Tong Y, Shi D. Federated latent Dirichlet allocation: A local differential privacy based framework. In: Proc. of the AAAI Conf. on Artificial Intelligence. AAAI, 2020. 6283−6290.

[86] Kairouz P, Oh S, Viswanath P. Extremal mechanisms for local differential privacy. In: Advances in Neural Information Processing Systems. MIT Press, 2014. 2879−2887.

[87] Liu R, Cao Y, Yoshikawa M, Chen H. Fedsel: Federated SGD under local differential privacy with top-*k* dimension selection. In: Proc. of the Int'l Conf. on Database Systems for Advanced Applications. Springer, 2020. 485−501.

[88] Shin H, Kim S, Shin J, Xiao X. Privacy enhanced matrix factorization for recommendation with local differential privacy. IEEE Trans. on Knowledge and Data Engineering, 2018, 30(9): 1770−1782.

[89] Agarwal N, Sures AT, Yu F, Kumar S, Mcmahan HB. cpSGD: Communication-efficient and differentially-private distributed SGD. In: Advances in Neural Information Processing Systems. MIT Press, 2018. 7564−7575.

[90] Suresh AT, Felix XY, Kumar S, McMahan HB. Distributed mean estimation with limited communication. In: Proc. of the Int'l Conf. on Machine Learning. ACM, 2017. 3329−3337.

[91] Aono Y, Hayashi T, Wang L, Moriai S. Privacy-preserving deep learning via additively homomorphic encryption. IEEE Trans. on Information Forensics and Security, 2017, 13(5): 1333−1345.

[92] Liu Y, Kang Y, Xing C, Chen T, Yang Q. A secure federated transfer learning framework. IEEE Intelligent Systems, 2020, 35(4): 70−82.

[93] Bonawitz K, Ivanov V, Kreuter B, Marcedone A, McMahan HB, Patel S, Seth K. Practical secure aggregation for privacy-preserving machine learning. In: Proc. of the 2017 ACM SIGSAC Conf. on Computer and Communications Security. ACM, 2017. 1175−1191.

[94] Heikkilä M, Lagerspetz E, Kaski S, Shimizu K, Tarkoma S, Honkela A. Differentially private bayesian learning on distributed data. In: Advances in Neural Information Processing Systems. MIT Press, 2017. 3226−3235.

[95] Zhang X, Ji S, Wang H, Wang T. Private, yet practical, multiparty deep learning. In: Proc. of the 2017 IEEE 37th Int'l Conf. on Distributed Computing Systems (ICDCS). IEEE, 2017. 1442−1452.

[96] Xu G, Li H, Liu S, Yang K, Lin X. Verifynet: Secure and verifiable federated learning. IEEE Trans. on Information Forensics and Security, 2019, 15: 911−926.

[97] Avent B, Korolova A, Zeber D, Hovden T, Livshits B. {BLENDER}: Enabling local search with a hybrid differential privacy model. In: Proc. of the 26th USENIX Security Symp. (USENIX Security 2017). USENIX Association, 2017. 747−764.

[98] Zhao L, Wang Q, Zou Q, Zhang Y, Chen Y. Privacy-preserving collaborative deep learning with unreliable participants. IEEE Trans. on Information Forensics and Security, 2019, 15: 1486−1500.

[99] Pathak MA, Rane S, Raj B. Multiparty differential privacy via aggregation of locally trained classifiers. In: Advances in Neural Information Processing Systems. MIT Press, 2010. 1876−1884.

[100] Jayaraman B, Wang L. Distributed learning without distress: Privacy-preserving empirical risk minimization. In: Advances in Neural Information Processing Systems. MIT press, 2018. 6346−6357.

[101] Truex S, Baracaldo N, Anwar A, Steinke T, Ludwig H, Zhang R, Zhou Y. A hybrid approach to privacy-preserving federated learning. In: Proc. of the 12th ACM Workshop on Artificial Intelligence and Security. ACM, 2019. 1−11.

[102] Paillier P. Public-key cryptosystems based on composite degree residuosity classes. In: Proc. of the Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Berlin, Heidelberg: Springer, 1999. 223−238.

[103] Damgård IB, Jurik MJ. A generalisation, a simplication and some applications of paillier's probabilistic public-key system. In: Proc. of the Int'l Workshop on Public Key Cryptography. Springer, 2001. 119−136.

[104] Xu R, 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. ACM, 2019. 13−23.

[105] Boneh D, Sahai A, Waters B. Functional encryption: Definitions and challenges. In: Proc. of the Theory of Cryptography Conf. Berlin, Heidelberg: Springer, 2011. 253−273.

[106] Kim M, Lee J, Ohno-Machado L, Jiang X. Secure and differentially private logistic regression for horizontally distributed data. IEEE Trans. on Information Forensics and Security, 2019, 15: 695−710.

[107] Zhao B, Fan K, Yang K, Wang Z, Li H, Yang Y. Anonymous and privacy-preserving federated learning with industrial big data. IEEE Trans. on Industrial Informatics, 2021, 17(9): 6314−6323.

[108] Hu R, Guo Y, Gong Y. Concentrated differentially private and utility preserving federated learning. arXiv preprint arXiv:2003.13761, 2020.

[109] Gong M, Feng J, Xie Y. Privacy-enhanced multi-party deep learning. Neural Networks, 2020, 121: 484−496.

[110] Choudhury O, Gkoulalas-Divanis A, Salonidis T, Sylla I, Park Y, Hsu G, Das A. Anonymizing data for privacy-preserving federated learning. arXiv preprint arXiv:2002.09096, 2020.

[111] Poulis G, Loukides G, Gkoulalas-Divanis A, Skiadopoulos S. Anonymizing data with relational and transaction attributes. In: Proc. of the Joint European Conf. on Machine Learning and Knowledge Discovery in Databases. Berlin, Heidelberg: Springer, 2013. 353−369.

[112] Bittau A, Erlingsson Ú, Maniatis P, Mironov I, Raghunathan A, Lie D, Seefeld B. Prochlo: Strong privacy for analytics in the crowd. In: Proc. of the 26th Symp. on Operating Systems Principles. ACM, 2017. 441−459.

[113] Erlingsson Ú, Feldman V, Mironov I, Raghunathan A, Song S, Talwar K, Thakurta A. Encode, shuffle, analyze privacy revisited: formalizations and empirical evaluation. arXiv preprint arXiv:2001.03618, 2020.

[114] Bassily R, Nissim K, Stemmer U, Thakurta A. Practical locally private heavy hitters. In: Advances in Neural Information Processing Systems. MIT Press, 2017. 2288−2296.

[115] Kwon AH, Lazar D, Devadas S, Ford B. Riffle: An efficient communication system with strong anonymity. Proc. on Privacy Enhancing Technologies, 2016, 2: 115−134.

[116] Erlingsson Ú, Feldman V, Mironov I, Raghunathan A, Talwar K, Thakurta A. Amplification by shuffling: From local to central differential privacy via anonymity. In: Proc. of the 30th Annual ACM-SIAM Symp. on Discrete Algorithms. Society for Industrial and Applied Mathematics, 2019. 2468−2479.

[117] Kasiviswanathan SP, Lee HK, Nissim K, Raskhodnikova S, Smith A. What can we learn privately? SIAM Journal on Computing, 2011, 40(3): 793−826.

[118] Ghazi B, Pagh R, Velingker A. Scalable and differentially private distributed aggregation in the shuffled model. arXiv preprint arXiv:1906.08320, 2019.

[119] Ishai Y, Kushilevitz E, Ostrovsky R, Sahai A. Cryptography from anonymity. In: Proc. of the 2006 47th Annual IEEE Symp. on Foundations of Computer Science (FOCS 2006). IEEE, 2006. 239−248.

[120] Ghazi B, Manurangsi P, Pagh R, Velingker A. Private aggregation from fewer anonymous messages. In: Proc. of the Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Springer, 2020. 798−827.

[121] Cheu A, Smith A, Ullman J, Zeber D, Zhilyaev M. Distributed differential privacy via shuffling. In: Proc. of the Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Cham: Springer, 2019. 375−403.

[122] Balle B, Bell J, Gascón A, Nissim K. The privacy blanket of the shuffle model. In: Proc. of the Annual Int'l Cryptology Conf. Cham: Springer, 2019. 638−667.

[123] Balle B, Bell J, Gascón A, Nissim K. Private summation in the multi-message shuffle model. In: Proc. of the 2020 ACM SIGSAC Conf. on Computer and Communications Security. ACM, 2020. 657−676.

[124] Ghazi B, Golowich N, Kumar R, Manurangsi P, Pagh R, Velingker A. Pure differentially private summation from anonymous messages. arXiv preprint arXiv:2002.01919, 2020.

[125] Ghazi B, Golowich N, Kumar R, Pagh R, Velingker A. On the power of multiple anonymous messages: Frequency estimation and selection in the shuffle model of differential privacy. In: Proc. of the Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Springer, 2021. 463−488.

[126] Liu R, Cao Y, Chen H, Guo R, Yoshikawa M. FLAME: Differentially private federated learning in the shuffle model. In: Proc. of the AAAI Conf. on Artificial Intelligence. AAAI, 2021. 8688−8696.

[127] Balle B, Barthe G, Gaboardi M. Privacy amplification by subsampling: Tight analyses via couplings and divergences. In: Advances in Neural Information Processing Systems. MIT Press, 2018. 6277−6287.

[128] Osia SA, Taheri A, Shamsabadi AS, Katevas K, Haddadi H, Rabiee HR. Deep private-feature extraction. IEEE Trans. on Knowledge and Data Engineering, 2018, 32(1): 54−66.

[129] Song L, Shokri R, Mittal P. Membership inference attacks against adversarially robust deep learning models. In: Proc. of the 2019 IEEE Security and Privacy Workshops (SPW). IEEE, 2019. 50−56.

[130] Lyu L, Yu J, Nandakumar K, Li Y, Ma X, Jin J, Ng KS. Towards fair and privacy-preserving federated deep models. IEEE Trans. on Parallel and Distributed Systems, 2020, 31(11): 2524−2541.

[131] Wu Y, Cai S, Xiao X, Chen G, Ooi BC. Privacy preserving vertical federated learning for tree-based models. arXiv preprint arXiv:2008.06170, 2020.

[132] Chen T, Jin X, Sun Y, Yin W. VAFL: A method of vertical asynchronous federated learning. arXiv preprint arXiv:2007.06081, 2020.

[133] Gao D, Liu Y, Huang A, Ju C, Yu H, Yang Q. Privacy-preserving heterogeneous federated transfer learning. In: Proc. of the 2019 IEEE Int'l Conf. on Big Data (Big Data). IEEE, 2019. 2552−2559.

附中文参考文献:

[5] 刘睿瑄, 陈红, 郭若杨, 赵丹, 梁文娟, 李翠平. 机器学习中的隐私攻击与防御. 软件学报, 2020, 31(3): 866−892. http://www.jos.org.cn/1000-9825/5904.htm [doi: 10.13328/j.cnki.jos.005904]

[7] 王健宗, 孔令炜, 黄章成, 陈霖捷, 刘懿, 卢春曦, 肖京. 联邦学习隐私保护研究进展. 大数据, 2021, 7(3): 130−149. http://kns.cnki.net/kcms/detail/10.1321.G2.20210112.1017.002.html

[38] 叶青青, 孟小峰, 朱敏杰, 霍峥. 本地化差分隐私研究综述. 软件学报, 2018, 29(7): 1981−2005. http://www.jos.org.cn/1000-9825/5364.htm [doi: 10.13328/j.cnki.jos.005364]

刘艺璇(1993－), 女, 博士生, 主要研究领域为机器学习, 联邦学习中的隐私保护.

刘宇涵(1996－), 女, 博士生, 主要研究领域为图数据的隐私保护.

陈红(1965－), 女, 博士, 教授, 博士生导师, CCF 杰出会员, 主要研究领域为大数据管理, 隐私保护.

李翠平(1971－), 女, 博士, 教授, 博士生导师, CCF 杰出会员, 主要研究领域为社交网络分析, 社会推荐, 大数据分析及挖掘.