联邦遗忘学习隐私安全与算法效率研究综述

唐湘云[1)] 王 伟[1)] 翁 彧[1)] 沈 蒙[2)] 张 焘[3)] 王 伟[4)] 祝烈煌[2)]

[1)](中央民族大学信息工程学院 北京 100081)

[2)](北京理工大学网络空间安全学院 北京 100081)

[3)](北京交通大学网络空间安全学院(国家保密学院) 北京 100044)

[4)](西安交通大学智能网络与网络安全教育部重点实验室 西安 710049)

摘 要 在数据驱动的人工智能应用迅猛发展的背景下,用户对其个人数据安全与隐私保护的需求持续提升。联邦学习作为一种分布式机器学习范式,通过共享模型参数而非原始数据来完成模型训练,缓解了用户数据的隐私泄露风险。然而,联邦学习仍难以满足用户从已训练模型中删除其个人数据的需求。为此,联邦遗忘学习被提出,旨在响应用户发起的数据遗忘请求,以擦除其数据对模型的影响,同时保持模型的有效性。但目前的联邦遗忘学习技术还存在隐私泄露隐患和数据安全问题,以及模型恢复开销过高的威胁。为深入探讨联邦遗忘学习技术在隐私安全和算法效率方面的研究现状,本文首先系统性地介绍了联邦遗忘学习的基本概念,并揭示了其所面临的隐私泄露风险高、模型性能难恢复、计算开销大和存储成本高四大核心挑战。随后,从隐私保护、模型恢复、计算效率和存储效率四个方面,详尽综述了联邦遗忘学习的研究进展,对相关方案进行了清晰的分类及对比总结,为后续研究提供了明确的理论与实践指导。最后,本文总结了联邦遗忘学习的实际应用,并对未来的研究方向进行了展望,以促进联邦遗忘学习在人工智能领域中的安全应用。

关键词 联邦学习;联邦遗忘学习;隐私保护;人工智能安全;隐私攻击

中图法分类号 TP18 **DOI号** 10.11897/SP.J.1016.2025.02064

A Survey on Privacy Security and Computation Efficiency in Federated Unlearning

TANG Xiang-Yun[1)] WANG Wei[1)] WENG Yu[1)] SHEN Meng[2)]
ZHANG Tao[3)] WANG Wei[4)] ZHU Lie-Huang[2)]

[1)](School of Information Engineering, Minzu University of China, Beijing 100081)

[2)](School of Cyberspace Science and Technology, Beijing Institute of Technology, Beijing 100081)

[3)](School of Cyberspace Science and Technology, Beijing Jiaotong University, Beijing 100044)

[4)](Ministry of Education Key Lab For Intelligent Networks and Network Security, Xi'an Jiaotong University, Xi'an 710049)

Abstract With the rapid advancement of data-driven artificial intelligence applications, concerns regarding personal data security and privacy protection have intensified among users. Federated Learning (FL) effectively mitigates issues such as privacy breaches and insufficient computational resources in traditional centralized model training modes. While legislations such as the EU General Data Protection Regulation and the California Consumer Privacy Act require ensuring both

收稿日期:2024-11-06;在线发布日期:2025-05-08。本课题得到国家自然科学基金青年基金(62302539、62402029)、国家自然科学基金联合基金重点项目(U23A20304)、中国博士后科学基金(2024T170047,GZC20230223,2024M750165)资助。**唐湘云**,博士,副教授,中国计算机学会(CCF)会员,主要研究领域为人工智能安全、联邦学习、数据安全与隐私保护。E-mail:xiangyunt@muc.edu.cn。**王 伟**,硕士研究生,主要研究领域为联邦学习。**翁彧(通信作者)**,教授,博士生导师,主要研究领域为人工智能安全。E-mail:wengyu@muc.edu.cn。**沈 蒙**,教授,博士生导师,中国计算机学会(CCF)会员,主要研究领域为数据安全、人工智能安全、区块链安全。**张 焘**,博士,副教授,中国计算机学会(CCF)会员,主要研究领域为物联网安全、移动目标防御。**王 伟**,教授,博士生导师,中国计算机学会(CCF)杰出会员,主要研究领域为网络与系统安全、区块链及隐私计算理论与技术。**祝烈煌**,教授,博士生导师,中国计算机学会(CCF)会员,主要研究领域为密码算法及安全协议、区块链技术。

data security and participants "right to be forgotten", FL participants may request the forgetting of their own data contributions to protect their privacy, and FL systems may intend to delete contributions from malicious users to maintain robustness. Federated Unlearning (FU) technologies address these forgetting requests, ensuring utility restoration of the original model while performing forgetting operations on participants or data samples, which has become a hot topic in current research. However, existing methods still have shortcomings in ensuring privacy security and improving algorithm efficiency. This paper first systematically introduces the basic concepts of federated unlearning and highlights the four core challenges it faces: high risk of privacy leakage, difficulty in restoring model performance, high computational overhead, and high storage costs. Subsequently, it provides a comprehensive review of research progress in federated unlearning from four aspects: privacy protection, model recovery, computational efficiency, and storage efficiency. Furthermore, the paper categorizes and compares related approaches clearly to provide a solid theoretical and practical foundation for future research. Finally, the paper summarizes the practical applications of federated unlearning and discusses potential future research directions to promote its safe application in the field of artificial intelligence. Compared with the existing Federated Unlearning survey, this paper differs in five aspects: (1) This paper identifies the core threats in federated unlearning through a thorough analysis of existing literature and approaches. These threats include privacy leakage, difficulty in model recovery, high computational costs, and increased storage costs. Unlike other reviews that mainly provide an overview, this paper discusses the limitations of various approaches in tackling these challenges and examines their specific impacts on model utility, system security, and algorithm efficiency. (2) This paper systematically summarizes research on privacy protection in federated unlearning, analyzing the effectiveness, applicability, and limitations of various strategies from multiple perspectives. It delves into differential privacy, homomorphic encryption, and other protocols, highlighting their effectiveness in safeguarding user data privacy and addressing privacy leakage. (3) This paper explores how to restore model performance through measures such as clearing residual data, verifying forgetting results, and enhancing model robustness. By comparing various data removal and verification strategies in terms of their effectiveness in thoroughly eliminating residual forgotten data, preventing catastrophic federated unlearning, and ensuring model stability, it reveals their strengths and weaknesses. (4) This paper reviews and evaluates methods for improving the computational and storage efficiency of federated unlearning. It discusses specific techniques such as compression strategies, sharding mechanisms, and knowledge distillation, comparing their advantages and disadvantages regarding speed, computational complexity, and space complexity. (5) This paper summarizes practical application scenarios of federated unlearning found in the literature, including personalized recommendations, healthcare, digital twins, and ranking learning. By organizing and analyzing these scenarios, it demonstrates the applicability, effectiveness, and limitations of federated unlearning in real-world contexts, providing a foundation for scholars to further explore solutions in specific scenarios.

Keywords　federated learning; federated unlearning; privacy protection; artificial intelligence security; privacy attacks

1　引　言

传统的机器学习(Machine Learning,ML)方法通常需要将数据集中到一个中央服务器或云端进行模型训练。然而,这种数据集中化的方式存在隐私泄露和集中受限的问题。首先,参与者的个人数据可能包含敏感信息,例如医疗记录、金融数据等,将

这些数据集中存储可能存在隐私泄露或数据滥用的风险。其次,许多国家和地区对数据隐私有严格的法律法规,例如欧盟通用数据保护条例(GDPR)[1]和加州消费者隐私法案(CCPA)[2],限制数据的跨境传输和集中存储,使得数据难以自由流动。此外,集中式存储还可能受到计算资源瓶颈、网络带宽限制和单点故障等问题的制约。例如,在大规模数据集的情况下,中央服务器的计算能力可能难以满足高效训练需求,同时,大量数据的远程传输会增加通信成本,导致训练效率下降。因此,数据的集中化不仅受法律法规的约束,还面临技术和资源层面的挑战。

由谷歌提出的联邦学习[3](Federated Learning,FL)已经成为解决传统机器学习中所存在隐私泄露和集中受限问题的主流方法[4]。参与者不需要共享原始私有数据,而是各自在本地训练模型。中央服务器也不需要对数据集中存储,而是接收参与者上传的本地训练模型的参数从而进行全局模型的聚合,有效地缓解了参与者的数据隐私泄露问题[5]。近年,联邦学习取得了显著的发展,并在医疗健康、金融服务和智能交通等多个领域得到了广泛应用。

然而,在现实的联邦学习应用中,“被遗忘权”需求愈发突出,即需要从已训练的全局模型中删除客户端或者数据样本的贡献。在隐私方面,一系列的数据安全立法,例如欧盟通用数据保护条例和 加州消费者隐私法案,为参与者提供了“被遗忘权”,参与者有权利要求公司或企业删除其隐私数据和贡献。在安全方面,当某些客户端在训练过程中有意或无意地使用低质量数据时,或其本身就是恶意客户端,即在训练过程中故意提交异常或篡改的数据,以加害全局模型的安全性[6]。删除这些客户端的贡献可以有效地提升联邦学习系统的鲁棒性和安全性。

机器遗忘学习(Machine Unlearning,MU)[7-9]技术提供了从全局模型中进行客户端或数据样本贡献删除的遗忘技术,其主要依赖于修改模型的权重来确保对于给定数据点的“遗忘”,即获得一个与假设未使用该数据点进行训练的模型等效的模型[10-12]。然而,机器遗忘学习无法直接应用于联邦学习场景中进行客户端或数据样本遗忘。在联邦学习中,不同参与方的数据分布和特征各异,直接应用机器遗忘学习可能会导致全局模型性能大幅下降。此外,全局模型通过聚合各个参与者的本地模型生成,直接应用机器遗忘学习会增加聚合过程的复杂性,且难以准确删除特定客户端的贡献,进而对其他参与者的训练产生负面影响。

因此,联邦遗忘学习[13](Federated Unlearning,FU)被提出,通过擦除目标客户端或样本数据的贡献来满足联邦学习中的遗忘请求,同时确保模型效用接近从头开始的效果。然而,现有的联邦遗忘学习方法面临遗忘过程中的隐私泄露风险及算法效率低下等问题。隐私泄露的隐患会导致整体系统的安全性降低,而低效的算法效率则会影响模型的性能恢复。针对上述问题,我们总结了联邦遗忘学习方案面临的主要挑战,包括隐私泄露风险高、模型性能难恢复、计算开销大和存储成本高。

(1)隐私泄露风险高。联邦遗忘学习中的隐私泄露问题可以从不同的遗忘者角度分为两类:当服务器作为遗忘者时,会保存大量的客户端历史信息,恶意服务器可能会访问客户端的历史更新梯度等信息从而导致隐私泄露;当目标客户端作为遗忘者时,在主动遗忘数据后会上传遗忘后的本地模型,服务器可以获取到目标客户端遗忘前后的两版模型。恶意服务器则会通过分析其遗忘前后的模型差异,从而进行成员推理和数据重构攻击导致目标客户端隐私泄露。

(2)模型性能难恢复。联邦遗忘学习中关于模型性能恢复的问题主要可以归结为以下两类:一类是遗忘数据的残留影响,例如采用近似恢复方法或缺乏严格的遗忘验证时,部分遗忘数据仍然会对模型产生影响。这种残留信息可能会导致模型在更新后仍然受遗忘数据的干扰,从而影响其性能恢复,使模型表现出不稳定的泛化能力或退化的推理能力;另一类是灾难性的联邦遗忘学习。若删除的数据包含关键信息(如高权重特征或决定性样本),可能导致模型性能大幅下降,甚至影响其推理能力。尤其是在移除对模型决策至关重要的数据时,模型可能无法通过后续训练完全恢复其原始性能,凸显了模型性能难恢复的严峻挑战。

(3)计算开销大。由于联邦学习是一种分布式的机器学习方法,模型是在多个客户端的数据基础上联合训练的,因此客户端的数据对模型的权重和性能产生了影响。当客户端发出遗忘请求时,为了确保模型不再包含该客户端的数据贡献,最直接的方法是从头开始重新训练整个模型。即在移除目标客户端的数据后,重新进行训练。然而,从头训练需要全部客户端参与训练,从而造成计算成本急剧增加。并且,如果系统频繁地收到客户端的遗忘请求,不断地重训练所需要的计算和时间开销将难以估计。

(4)存储成本高。在联邦遗忘学习中,存储效率对于系统的正常运行也至关重要。在进行遗忘和恢复全局模型的过程中,服务器会保留大量的历史信息,包括历史梯度和模型更新等信息。此外,客户端有时也需要保存模型的中间状态或更新记录,以便在遗忘请求发生时能够协助删除目标数据的影响。这将导致存储成本急剧升高,从而增加在资源受限环境中部署和运行模型的难度,同时也可能导致参与者因无法接受过高的存储成本而退出系统训练。

为了应对上述挑战,我们将从隐私保护、模型恢复、计算效率和存储效率四个方面总结和讨论现有的联邦遗忘学习方案。

(1)隐私保护。针对服务器执行遗忘时所造成的隐私泄露问题,主要是利用差分隐私的方法对客户端上传的信息进行保护,或者直接避免利用客户端的历史信息实现遗忘效果,从而实现对联邦遗忘学习的隐私保护。此外,解决客户端执行遗忘所造成的隐私泄露问题时,常将同态加密、差分隐私等技术集成到联邦遗忘学习算法中,抵御成员推理、数据重构等隐私攻击,从而保护训练数据的隐私。

(2)模型恢复。针对遗忘数据残留威胁模型性能的问题,现有方案主要通过遗忘后的进一步验证方式,或者是利用懒惰学习策略进一步清除,目的是确保遗忘数据被彻底移除,以防其在系统中持续存在而影响模型性能的恢复;针对灾难性联邦遗忘所造成的模型可用性与稳定性降低问题,通常采用可塑权重巩固策略弹性地约束权重变化,或使用动态惩罚策略惩罚与触发数据集和触发模式无关信息的过度遗忘学习,以此来避免灾难性联邦遗忘现象,从而确保模型的可用性。

(3)计算效率。在联邦遗忘学习中,如何高效地实现“遗忘”是亟待解决的难题。针对联邦遗忘学习中计算开销大的问题,现有方案主要有两类:一类是通过利用保存的客户端历史信息进行恢复全局模型,在中央服务器或是目标客户端执行遗忘操作时,通过适当放宽一定的模型精度,使用历史信息近似出遗忘后的全局模型,从而显著提高计算效率;另一类是通过快速重新训练进行恢复全局模型,从而彻底遗忘目标数据的影响,这类方案旨在保证模型精度的前提下提升计算效率。

(4)存储效率。在执行遗忘操作时,服务器通常需要存储大量客户端的历史信息,有时客户端本地也需要存储额外的信息。因此,提升存储效率至关重要。可以通过记录并利用过往的模型参数和更新,在遗忘或回滚操作中加速遗忘后的重建。此外,还可以通过压缩、优化或者选择性存储等方法减少对历史数据的依赖,这通常侧重于在每轮训练中动态优化参数更新,避免长期存储大量历史数据,从而提升存储效率。

现有的联邦遗忘学习综述文献主要集中在基本概念和方法分类上,缺乏对联邦遗忘学习具体算法的深入分析以及算法性能的对比[14-15]、忽略了联邦遗忘学习隐私与安全威胁的系统讨论[16]、缺少现有方案效率的可靠总结[15,17-18],且尚未有综述对联邦遗忘学习的应用场景进行分析与总结。

相比已有文献,除了系统性地介绍联邦遗忘学习基本概念、方法分类以及算法评估指标之外,我们的联邦遗忘学习综述还具有以下主要创新之处:

(1)揭示联邦遗忘学习面临的核心挑战。通过对现有文献的深入分析和对联邦遗忘学习方案的系统梳理,本综述揭示了联邦遗忘学习面临的核心挑战,包括隐私泄露、模型恢复、计算开销大、存储成本高。与现有综述主要停留在问题概述层面不同,本综述通过对不同方案在应对这些挑战时的局限性进行讨论,深入探讨了这些挑战对系统安全性、模型效用及算法效率的具体影响,旨在为后续研究提供明确的方向和针对性的解决思路。

(2)系统解析隐私保护策略的实现与挑战。本综述系统归纳了联邦遗忘学习中隐私保护相关的研究,并从多角度详细分析了各类隐私保护策略的有效性、适用性及其局限性。通过对差分隐私、同态加密及其他隐私保护协议的深入讨论,揭示了不同方法在保护联邦遗忘学习中用户数据隐私时的具体实现效果以及应对隐私泄露的能力。

(3)详尽探讨确保模型性能恢复的方案及效果。本综述分析了如何通过清除残留遗忘数据、验证遗忘结果、增强模型鲁棒性等手段保障模型的可用性。通过对各类遗忘策略在应对残留遗忘数据、避免灾难性遗忘、提升模型稳定性等方面的表现进行了可靠的对比分析,综述揭示了各类遗忘策略的关键技术以及在应对模型性能恢复困境的优劣势。

(4)评估计算与存储效率的优化策略。本综述对提升联邦遗忘学习计算效率和存储效率的方法进行了系统性梳理和对比评估,探讨了压缩策略、分片机制、知识蒸馏等具体实现技术。通过加速比、计算复杂度、空间复杂度等方面的详细比较,本综述揭示了不同方法在提升计算效率和存储效率时的优缺

点,可为研究人员在选择和优化高效的联邦遗忘学习方案时提供参考。

(5)梳理联邦遗忘学习已有的实际应用场景。不同于现有综述大多侧重理论讨论,本综述归纳总结了现有文献中提及的联邦遗忘学习实际应用场景,包括个性推荐、医疗健康、数字孪生和排名学习等。通过对应用场景的梳理和性能表现的具体分析,展示出联邦遗忘学习在实际场景中的适用性、效果和局限性,为联邦遗忘学习在不同行业中的实施提供案例,为研究者探索和优化在特定场景下的联邦遗忘学习方案提供研究基础,表 1 将本文与现有联邦遗忘学习综述进行了比较。

表 1 联邦遗忘学习综述比较

年份	文献	遗忘分类定义	遗忘评估指标	隐私安全分析	计算效率对比	存储效率分析	相关应用介绍
2023	[14]	◐	◐	◐	◎	◐	◎
2023	[15]	◎	◎	●	◎	◎	◎
2023	[16]	◐	◐	◐	◎	◎	◐
2024	[17]	◐	●	◐	◎	◎	◎
2024	[18]	◐	◐	◐	◐	◐	◐
2024	本文	●	●	●	●	●	●

注:●代表具体描述、◐有描述但不全面、◎代表没有描述。

本文的组织结构安排如下:第 2 章介绍联邦遗忘学习基本概念和分类,并总结现有的关于联邦遗忘学习算法或框架的评估指标;第 3 章探讨联邦遗忘学习在隐私方面面临的挑战,通过服务器和客户端作为遗忘者的不同角度综述相关隐私保护方案;第 4 章在阐述联邦遗忘学习中的模型性能恢复后,从残留遗忘数据对模型性能的影响和灾难性遗忘对模型可用性的降低两个角度,综述相关遗忘方案;第 5 章通过基于历史信息遗忘和快速重新训练遗忘两种恢复全局模型的方式对提升计算效率方案进行综述,并且在计算效率方面将各个算法与重新训练和相关经典算法进行了对比;第 6 章根据是否依赖历史信息提升存储效率将相关方案分为两类进行综述,并将相关方案进行了对比分析;第 7 章从实际应用的层面对联邦遗忘学习的可用性进行了验证;最后,第 8 章对全文的研究工作进行了总结,并提出了未来可能的研究方向。图 1 展示了本文综述的组织结构。

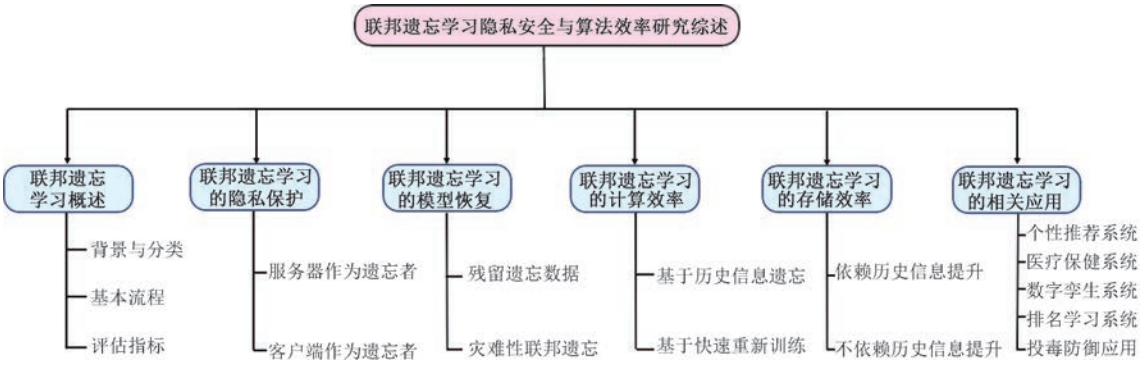

图 1 联邦遗忘学习综述的组织结构图

2 联邦遗忘学习概述

2.1 联邦遗忘学习背景与分类

在联邦遗忘学习中,参与者可以选择撤销或从模型中删除他们的数据贡献。这意味着即使参与者之前在训练期间向模型贡献了数据,他们也可以请求从模型中删除数据,而不会影响其他参与者的数据贡献[19]。联邦遗忘学习的过程需要更新全局模型,以适应参与者数据的删除。这可以通过重新训练模型,或使用特定技术调整模型参数来实现,从而确保即使删除特定参与者的数据后,模型仍能正常运行,保持准确性和有效性[20]。

联邦遗忘学习中涉及了服务器、目标参与者、其他参与者三类主要角色。这些角色在整个遗忘学习的过程中发挥着不可或缺的作用,各自负责特定的职责和功能。

(1)服务器。在联邦遗忘学习中,服务器发挥着协调和管理的作用。服务器负责收集来自各个参与者(例如设备或客户端)的本地模型更新,并将它们

训练成一个全局模型。在训练过程中,服务器将全局模型的参数发送给参与者,以便他们能够在本地训练数据上进行模型更新。除此之外,当客户端主动发起遗忘请求或者系统需要删除某些恶意数据时,服务器还可以进行遗忘操作实现贡献擦除的目的。

(2)目标参与者。在联邦遗忘学习中,当某些参与者的数据或者整个参与者需要被遗忘时,我们将其统称为目标参与者。目标参与者可以主动或被动地请求遗忘。在执行遗忘操作后,目标参与者的数据或其对联邦学习系统的贡献将被删除,且该参与者不再参加后续的模型训练,也不会对新的全局模型产生影响。

(3)其他参与者。在联邦遗忘学习中,将除目标参与者之外的剩余参与者称为其他参与者。如果目标参与者只需要遗忘部分数据贡献,遗忘操作后,其他参与者与目标参与者剩余的数据将共同用于训练新的全局模型;如果整个目标参与者的数据贡献都需要被遗忘,那么后续的训练将只由其他参与者与服务器共同来完成。

我们给出通用的符号定义:将服务器定义为 S,客户端定义为 $C=\{C_1,C_2,C_3,\cdots,C_k\}$,客户端的数据集定义为 $\mathcal{D}_{C_i}=\{x_i,y_i\}_{i=1}^{n_{C_i}}$($C_i$ 表示第 i 个客户端,n_{C_i} 表示样本数据量),服务器聚合客户端的本地模型所得的全局模型定义为 ω^*,客户端 C_i 的模型参数定义为 $\omega_n^{C_i}$(n 表示为训练轮数)。我们将 $\mathcal{D}$ 定义为总体数据集,$\mathcal{D}_r$ 定义为保留集,$\mathcal{D}_f$ 定义为遗忘集。客户端发出的遗忘请求定义为 $U_k=\{u_1,u_2,\cdots,u_k\}$。

定义 1.　联邦遗忘学习。联邦遗忘学习指的是通过一种遗忘算法,使得全局模型遗忘特定的样本或数据集,而无需直接访问这些数据。当满足以下两个不等式时,即代表在遗忘算法 U 的作用下,遗忘后模型的输出与通过保留集 $\mathcal{D}_r$ 后重新训练的模型的输出之间差异非常小,即实现了联邦遗忘学习。

$$\Pr[A(\mathcal{D}_r)\in R]\leqslant e^{\epsilon}\Pr[U(A(\mathcal{D}),\mathcal{D}_f,\mathcal{D})\in R]+\delta \tag{1}$$

$$\Pr[U(A(\mathcal{D}),\mathcal{D}_f,\mathcal{D})\in R]\leqslant e^{\epsilon}\Pr[A(\mathcal{D}_r)\in R]+\delta \tag{2}$$

其中,$A(\mathcal{D})$ 表示联邦学习算法在数据集 $\mathcal{D}$ 上训练得到的模型,$U(A(\mathcal{D}),\mathcal{D}_f,\mathcal{D})$ 表示经过遗忘算法 U 处理后得到的模型。R 表示模型输出的结果空间,ϵ 和 δ 则表示差分隐私中的隐私预算和容忍概率。

接下来,我们从不同的角度给出联邦遗忘学习的分类。从遗忘目标上分类可以分为遗忘样本级目标、遗忘类别级目标以及遗忘客户端级目标三种;从遗忘方式上可以分为利用历史信息遗忘方法和快速重新训练遗忘方法。

2.1.1　从遗忘目标分类

在联邦遗忘学习中,遗忘请求可能会涉及目标客户端的部分特定数据或整个客户端。根据遗忘目标的不同,可以设计不同的联邦遗忘算法从而实现最佳的性能与效果。在联邦遗忘学习中,通常根据遗忘目标的不同分成样本级遗忘、类别级遗忘和客户端级遗忘三种,我们给出不同级别的遗忘目标的明确定义如下:

(1)样本级联邦遗忘学习。在联邦遗忘学习中,当遗忘目标为数据样本[21]时,我们需要删除目标客户端的特定数据样本以此来完成遗忘请求,之后通过此目标客户端的剩余数据样本以及其他客户端共同训练出新的全局模型。我们给出对于样本级联邦遗忘学习的明确定义如下:

定义 2.　样本级联邦遗忘学习。对于样本级联邦遗忘学习来说,设目标客户端为 C_t,$\mathcal{D}_r^t$ 为目标客户端剩余数据样本,B 是最小删除批量,ΔB_t 是目标客户端为 C_t 的删除量变化,用于调整删除批量的大小。当目标客户端为 C_t 发出遗忘请求 u_t 时:

$$\mathcal{D}_r^t=\mathcal{D}_{C_t}\setminus\mathcal{D}_f \tag{3}$$

$$\mathcal{D}_r=\mathcal{D}_r^1\cup\mathcal{D}_r^2\cup\cdots\cup\mathcal{D}_r^t\cup\cdots\cup\mathcal{D}_r^k \tag{4}$$

$$F_c^t(\omega)=\frac{1}{B-\Delta B_t}\sum\nolimits_{(x_i,y_i)\in\mathcal{D}_r}\ell_i(\omega) \tag{5}$$

其中,$\ell_i(\omega)$ 表示样本 (x_i,y_i) 在模型参数 ω 下的损失。在验证遗忘效果时,通常利用 KL 散度来衡量遗忘后的全局模型参数 ω^* 与遗忘前的全局模型参数 ω 之间的差异,给定一个极小的非负实数阈值 ξ,当

$$KL(P(\omega)\,\|\,Q(\omega^*))\leqslant\xi \tag{6}$$

成立时,则代表遗忘学习算法 U 成功地实现了样本级联邦遗忘学习。其中 $P(\omega)$ 和 $Q(\omega^*)$ 分别代表的是遗忘前后全局模型的输出分布。

(2)类别级联邦遗忘学习。在联邦遗忘学习中,针对分类任务中的类别进行的遗忘学习[22]即为类别级联邦遗忘学习。这类数据通常会同时存在于多个客户端中,因此需要在这些客户端上删除目标类别级数据以此来达到遗忘的目的,在执行遗忘操作之后,这些客户端剩余的数据来参与共同训练从而

构建出新的全局模型。

定义 3. 类别级联邦遗忘学习。设多个目标客户端的集合为 $C_x=\{C_q,C_w,C_r,\cdots,C_p\}$，$\mathcal{D}_{C_x}$ 表示目标客户端集合 C_x 的数据样本总数，$\mathcal{D}_r^x$ 为目标客户端剩余数据样本总数，B 是最小删除批量，ΔB_x 是目标客户端集合 C_x 的删除量变化。当目标客户端 C_x 发出遗忘请求 u_x 时：

$$\mathcal{D}_r^x=\mathcal{D}_{C_x}\setminus\mathcal{D}_f \tag{7}$$

$$\mathcal{D}_r=\mathcal{D}_r^1\cup\mathcal{D}_r^2\cup\cdots\cup\mathcal{D}_r^x\cup\cdots\cup\mathcal{D}_r^k \tag{8}$$

$$F_c^x(\omega)=\frac{1}{B-\Delta B_x}\sum\nolimits_{(x_i,y_i)\in\mathcal{D}_r}\ell_i(\omega) \tag{9}$$

验证遗忘学习算法 U 成功实现了类别级联邦遗忘学习的方式与公式(6)相同。

(3)客户端级联邦遗忘学习。当遗忘目标为整个客户端的所有数据贡献时[23]，我们称为客户端级联邦遗忘学习。当目标客户端希望退出共同训练过程，但又不想保留数据和模型更新信息以确保隐私时，可以主动实现客户端级联邦遗忘学习；当目标客户端被检测为恶意客户端时，服务器为了维护系统安全，也会对该客户端执行遗忘操作，此时目标客户端进行被动的客户端级联邦遗忘学习。我们给出对于客户端级联邦遗忘学习的明确定义如下：

定义 4. 客户端级联邦遗忘学习。设目标客户端为 C_v，$\mathcal{D}_{C_v}$ 为目标客户端的本地数据集，B 是最小删除批量，ΔB_v 是目标客户端为 C_v 的删除量变化。当目标客户端 C_v 发出遗忘请求 u_v 时：

$$\mathcal{D}_r=\mathcal{D}\setminus\mathcal{D}_{C_v} \tag{10}$$

$$F_c^v(\omega)=\frac{1}{B-\Delta B_v}\sum\nolimits_{(x_i,y_i)\in\mathcal{D}_r}\ell_i(\omega) \tag{11}$$

验证遗忘学习算法 U 成功实现了客户端级联邦遗忘学习的方式与公式(6)相同。

我们在表 2 中总结了现有联邦遗忘学习算法的遗忘方式，以及这些算法是否可以实现样本级、类别级和客户端级的遗忘。

表 2 联邦遗忘学习算法中遗忘方式及实现遗忘目标分类

遗忘方式	算法名称	遗忘样本级目标	遗忘类别级目标	遗忘客户端级目标
利用历史信息恢复	FedEraser[24]	√	√	√
	FUCP[25]	×	√	√
	FUKD[26]	×	×	√
	VERIFI[27]	√	√	√
	SGA-EWC[28]	√	√	√
	FRU[29]	×	×	√
	FedLU[30]	√	√	√
	UBA[31]	√	√	√
	2F2L[32]	√	√	√
	FFMU[33]	√	√	√
	FedRecovery[34]	√	√	√
	SIFU[35]	×	×	√
	MoDe[36]	√	√	√
	QUICKDROP[37]	√	√	√
	Exact-Fun[38]	√	√	√
快速重新训练恢复	SE[39]	×	×	√
	BCETFU[40]	√	√	√
	Rapid Retraining[41]	√	√	√
	KNOT[42]	√	√	√
	FedAF[43]	√	√	√
	UPGA[44]	×	√	√

2.1.2 从遗忘方式分类

现有的联邦遗忘学习研究从不同的遗忘方式上可以分为两类。第一类是利用历史信息遗忘，这种方法通常需要服务器利用历史梯度和模型权重来有效地逼近遗忘过程中的梯度。在此过程中，服务器需要消耗大量存储空间来保存每轮客户端的更新信息，旨在通过降低对恢复模型有效性和高精度的要求，从而提高联邦遗忘的效率。

第二类是基于快速重新训练的遗忘方法，即在删除某些客户端的贡献或数据样本后，剩余的所有客户端或数据样本共同参与训练。在完成遗忘操作后，客户端和服务器都需要参与接下来的全局模型恢复过程，通过多轮迭代直至全局模型收敛。这种方法虽然需要较高的计算成本，但能够恢复与原有全局模型相近的高精度。两种不同典型的联邦遗忘方法对比如下表 3 所示。

表 3　两种不同典型的联邦遗忘方法对比

联邦遗忘方法	计算成本	存储成本	恢复模型准确率
利用历史信息遗忘[24]	较低	较高	较低
快速重新训练遗忘[41]	较高	较低	较高

2.2　联邦遗忘学习的基本流程

联邦遗忘学习通过遗忘目标客户端或其数据样本来解决联邦学习系统中的隐私安全问题。如图 2 所示,其步骤可以概括如下:

(1)初始化。服务器从所有客户端中选择一定

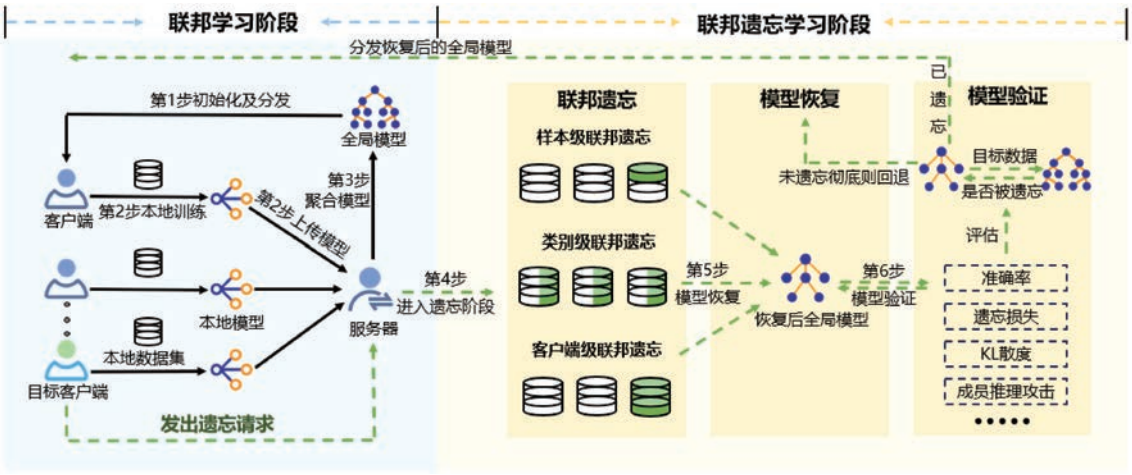

图 2　联邦遗忘学习基本流程图

比例的客户端参与联邦学习任务。服务器将初始化的全局模型 ω_0 广播给所有客户端,即 $\omega_0^C \leftarrow \omega_0$。

(2)本地训练。对于第 t 轮训练,每个客户端在其本地数据集 D_{C_i} 上训练接收到的全局模型 ω_t^c。在客户端,目标是最小化以下目标函数:

$$F_C(\omega)=\frac{1}{n_C}\sum_{i\in D_{C_i}}\ell_i(\omega) \tag{12}$$

对于给定的联邦学习任务,我们通常将

$$\ell_i(\omega)=\frac{1}{2}(x_i^T\omega-y_i) \tag{13}$$

作为客户端的损失函数,即用模型参数 ω 对训练示例 (x_i,y_i) 预测损失,然后每个客户端将其模型更新 $\Delta\omega_c=\eta\nabla F_c(\omega)$ 上传到服务器,其中 η 是学习率。

(3)聚合模型。服务器使用 FedAvg[45-46] 等模型更新聚合规则来聚合所有更新,以获得一个新的全局模型 ω_{t+1}。具体来说,在服务器端,目标是最小化以下目标函数:

$$f(\omega)=\sum_{C=1}^{C}\frac{n_C}{n}F_C(\omega) \tag{14}$$

$$F_C(\omega)=\frac{1}{n_C}\sum_{i\in D_{C_i}}\ell_i(\omega) \tag{15}$$

(4)遗忘阶段。当客户端请求数据删除时,服务器协调或执行数据删除操作。不同的遗忘目标的遗忘操作有所不同,具体定义细节详见 2.1.1 小结中的遗忘目标分类。

(5)模型恢复。在服务器完成数据删除操作后,需要对全局模型进行恢复。模型恢复的核心目标是消除因遗忘特定数据而对模型造成的影响,并确保模型在训练过程中不会再利用已删除数据的特征。恢复的方式可以分为基于历史信息恢复和快速重新训练恢复,即在不影响整体性能的情况下,确保被要求遗忘的数据不会在模型中残留信息。

(6)模型验证。模型恢复后,需要进行严格的模型验证,以确保遗忘过程的有效性。模型验证主要包括两方面:一是检查模型是否仍然保持较高的整体性能,二是验证被遗忘数据的影响是否已有效消除。通常,验证过程会使用不同的数据集,包括保留集和遗忘集,通过准确率、遗忘损失、KL 散度或成员推理攻击来评估模型的泛化能力和遗忘效果。若验证结果表明模型在保留集上的性能保持稳定,同时在遗忘集上已有效降低其对被遗忘数据的依赖,则可判定模型成功实现了预期的遗忘目标,成功实现了联邦遗忘学习。

2.3　联邦遗忘学习的评估指标

评估指标用于衡量研究所关注的变量或现象,从而验证研究的有效性和可靠性。通过合适的评估指标,可以客观地评估联邦遗忘学习算法的特征、遗忘效果和模型性能的表现。我们调查并总结了关于 FU 的评估指标,将其分为模型性能指标和恢复效率指标两个方面,以帮助读者更深入地了解联邦遗忘学习的各个特征。

2.3.1 模型性能指标

在联邦遗忘学习中，进行遗忘操作后恢复出的全局模型性能表明是否能够近似于从头开始训练所得到的全局模型。模型性能指标包括以下方面：

(1)全局模型恢复率：即在进行遗忘操作后恢复出的全局模型与重新训练恢复出的全局模型的差异。从遗忘的角度来看，这也可以称为遗忘操作的删除能力。数据样本被删除得越彻底，恢复出的全局模型与原模型的性能差异就越小。通过后门攻击成功率(Backdoor Attack Success Rate，BASR)[47-48]可以评估模型在应对后门攻击时的稳定性和泛化性能，从而分析联邦遗忘学习抵御此攻击的能力。成员推理攻击(Membership Inference Attack，MIA)[49-50]用于衡量模型在遭受攻击时的表现，以进一步评估联邦遗忘学习所恢复出的全局模型的鲁棒性。

(2)遗忘损失：在删除目标数据后，模型在剩余数据集上预测精度与稳定性下降的损失。遗忘损失值越小则代表恢复模型效用越高。

(3)准确率(遗忘集)：遗忘集指的是需要被遗忘的数据，即用户请求删除的样本。模型在遗忘集上的准确率直接反映了遗忘数据的影响是否被有效去除。如果遗忘成功，模型在这些数据上的准确率应显著降低，甚至接近随机猜测的水平。如果遗忘后的模型在遗忘集上仍然保持较高的准确率，则说明模型仍然保留了一定程度的遗忘数据信息，遗忘效果可能不够彻底。

(4)准确率(保留集)：保留集包含未受遗忘影响的数据，它用于衡量模型整体的泛化能力是否受到遗忘操作的影响。理想情况下，遗忘过程应仅影响遗忘数据，而不会削弱模型对保留数据的学习效果。因此，模型在保留集上的准确率应尽可能保持稳定。如果保留集的准确率下降明显，则说明遗忘方法可能对模型结构或参数造成了较大干扰，影响了模型的整体性能。

(5)遗忘率(Forget Rate，FR)[51]：FR 可以用来度量遗忘的性能，其计算公式为

$$FR=\frac{AF-BF}{BT}\times 100\% \tag{16}$$

遗忘率的计算需要借助成员关系判决器的预测。该判决器使用已知成员(在模型的训练集中)和非成员(不在训练集中)的样本训练一个二分类器。训练集中数据点的输出通常显示较高的置信度，而非训练集的数据点则通常具有较低的置信度或较大的不确定性。因此，这个分类器可以根据目标模型的输出(例如，预测概率分布、损失值等)来判断样本是否是训练集中的成员。

我们考虑的仅是遗忘前后成员为真实的情况：成员在联邦遗忘学习之前，BT 代表由成员关系判决器预测为真的样本数量，而 BF 代表由成员关系判决器预测为假的样本数量；经过联邦遗忘学习后，AT 代表由成员关系判决器预测为真的样本数量，而 AF 代表由成员关系判决器预测为假的样本数量。遗忘率的混淆矩阵如下表 4 所示，FR 越高则代表遗忘效果越好，即当 FR 达到 100%时代表将遗忘目标全部遗忘。

表 4 遗忘率混淆矩阵

	预测为成员(1)	预测为非成员(0)
真实为成员(1)	BT(遗忘前) AT(遗忘后)	BF(遗忘前) AF(遗忘后)
真实为非成员(0)	—	—

(6)对称绝对百分比误差(Symmetric Absolute Percentage Error，SAPE)[41]：SAPE 可以当作遗忘性能的指标，其计算公式为

$$\begin{aligned}\varepsilon_s &= SAPE(Acc^{*}_{\text{test}}, Acc^{u}_{\text{test}})\\ &=\frac{|Acc^{u}_{\text{test}}-Acc^{*}_{\text{test}}|}{|Acc^{*}_{\text{test}}|+|Acc^{u}_{\text{test}}|}\end{aligned} \tag{17}$$

其中，Acc^{*}_{test} 表示基线算法在测试数据集 D_{test} 上获得的模型 w^{*} 的准确性，Acc^{u}_{test} 表示所提出算法在同一数据集上获得的模型 w^{u} 的准确性。较低的 SAPE 值有助于保持模型效用，确保模型的性能不会因数据删除而受到显著影响。

(7)重构误差(Reconstruction Error，RE)[52]：RE 可以衡量模型重建它遗忘学习数据的程度，分数越低越好。其计算公式为

$$\begin{aligned}RE &= \|\boldsymbol{X}-\hat{\boldsymbol{X}}\|_F\\ &=\sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}(x_{ij}-\hat{x}_{ij})^2}\end{aligned} \tag{18}$$

其中，$\|\cdot\|_F$ 表示弗罗贝尼乌斯范数，$\boldsymbol{X}$ 表示原始数据矩阵，$\hat{\boldsymbol{X}}$ 表示重构后数据矩阵。其中，x_{ij} 是原始数据矩阵 $\boldsymbol{X}$ 中的元素，$\hat{x}_{ij}$ 是重构数据矩阵 $\hat{\boldsymbol{X}}$ 中的对应元素。

(8)激活距离(Activation Distance，AD)[53]：AD 可以测量模型在一组特定的遗忘数据上使用称为 L2 距离的模型在遗忘前后的预测之间的平均距离。其计算公式为

$$AD(\boldsymbol{a}_1,\boldsymbol{a}_2)=\|\boldsymbol{a}_1-\boldsymbol{a}_2\|_2$$

$$=\sqrt{\sum_{i=1}^{n}(a_{1i}-a_{2i})^2} \tag{19}$$

其中，$\boldsymbol{a}_1$,$\boldsymbol{a}_2$ 为两个不同输入 $\boldsymbol{x}_1$,$\boldsymbol{x}_2$ 分别经过某一层的网络输出，$\|\cdot\|_2$ 表示欧几里得范数，a_{1i} 和 a_{2i} 分别表示激活向量 $\boldsymbol{a}_1$ 和 $\boldsymbol{a}_2$ 的第 i 个元素，n 是激活向量的维度。

(9)KL(Kullback-Leibler)散度[54]:具体来说，KL 散度是一种度量两个概率分布之间相似性的方法。其计算公式为

$$D_{KL}(P\|Q)=\sum_i P(i)\log\frac{P(i)}{Q(i)} \tag{20}$$

其中，$P(i)$ 是分布 P 在事件 i 上的概率，$Q(i)$ 是分布 Q 在事件 i 上的概率。它总是正且在 $P(i)=Q(i)$ 时为零，这意味着最小化遗忘拉格朗日中的 KL 散度项可以使得遗忘后的模型权重分布与重新训练的数据的模型权重分布相似，从而实现遗忘的目标。

(10)影响函数 (Influence Function,IF)[27]:用于衡量某个数据点在模型上的影响，并评估遗忘操作是否成功。如果某个数据点 z 在遗忘后对模型 ω^* 的影响函数趋近于零，则说明该数据已成功被遗忘。其计算公式为

$$IF(z)=-\boldsymbol{H}_{\omega^*}^{-1}\nabla_{\omega}L(\omega^*,z) \tag{21}$$

其中，ω^* 是遗忘后的全局模型，$\boldsymbol{H}_{\omega^*}$ 是遗忘后模型的 Hessian 矩阵，用来衡量损失函数的曲率。$\nabla_{\omega}L(\omega^*,z)$ 是指损失函数对参数的梯度，计算数据点 z 在该模型中的影响。

(11) L_2 范数[41]:即欧几里得范数，可以用来量化遗忘前后模型的预测分布差异，从而判断目标数据的影响是否已消除和遗忘是否影响了模型的整体性能。其计算公式为

$$||\Delta P(y\mid x)||_2=||P_{\omega}(y\mid x)-P_{\omega^*}(y\mid x)||_2 \tag{22}$$

其中，$P_{\omega}(y\mid x)$ 代表遗忘前模型 ω 对测试数据 x 的预测概率分布，而 $P_{\omega^*}(y\mid x)$ 则代表遗忘后模型 ω^* 对测试数据 x 的预测概率分布。当 $||\Delta P(y\mid x)||_2\approx 0$ 时，则表示遗忘后的模型仍然与原始模型相似，说明模型在测试数据上的输出分布基本未受影响，遗忘可能成功且未影响整体性能。

(12)预期校准误差(Expected Calibration Error,ECE)[55]:

$$ECE=\sum_{m=1}^{M}\frac{|B_m|}{n}|Acc(B_m)-conf(B_m)| \tag{23}$$

其中，$Acc(B_m)$ 为第 m 个区间 B_m 内样本的平均准确度，$conf(B_m)$ 则为平均置信度，$|B_m|/n$ 则是第 m 个区间内样本占总样本数的比例。ECE 可以用于评估概率模型的校准，重点关注预测概率与实际观察频率之间的对齐并计算平均绝对误差从而来评估模型的校准性。ECE 越低，表示模型的预测概率与实际观测概率越接近，模型的校准性越好。

模型性能指标可以评估联邦遗忘学习算法恢复全局模型的能力和擦除目标数据贡献的能力。不同的遗忘优化算法应当根据遗忘前后模型的差异预测和恢复后全局模型抵御攻击的效果选择相对应的指标来进行评估，从而验证其方案在模型性能方面的提升。

2.3.2　遗忘效率指标

在实际的联邦遗忘学习环境中，有限的计算资源使得我们不仅要关注恢复全局模型的效用，遗忘效率的提升也至关重要。遗忘效率指标如下：

(1)收敛轮数:指在训练过程中模型参数不再显著变化或损失函数不再显著下降所需的迭代轮数。在联邦遗忘学习中，主要是指在遗忘操作后，所恢复出与遗忘前相当效用的全局模型所需要的轮数。

(2)遗忘时间:指发出遗忘请求后，从执行遗忘操作删除目标数据到恢复出与之前相似性能水平的全局模型所需要的运行时间，通过恢复时间可以直接反映出算法的遗忘效率。

(3)加速比:作为一个重要的评估指标去反映提出算法与从头开始训练算法或相关经典算法相比的效率提升。

$$r=\frac{T_{\text{Baseline}}}{T_{\text{New}}} \tag{24}$$

其中，r 代表加速比，T_{Baseline} 代表基线算法的遗忘时间，T_{New} 表示所提出的新算法的遗忘时间。

加速比越大，表明新算法相比于基线算法的遗忘效率提升越显著。

(4)存储效率:指系统在存储数据时所占用的空间大小与存储数据量之间的比例关系。在联邦遗忘学习恢复全局模型的过程中，无论是使用快速重训练还是微调的方式，都需要存储大量的模型参数和更新数据信息。特别是对于资源受限的设备和参与方，存储效率的快慢也会极大程度地影响遗忘的效率。

(5)通信效率:在联邦遗忘学习中，通信效率主要是指执行遗忘操作后，客户端与服务器需要进行大量的通信来恢复全局模型，在此过程中所需的通

信资源(如时间和带宽)与成功传输的有效数据量之间的关系。

$$\text{通信效率}=\frac{\text{有效传输数据量}}{\text{总传输数据量}} \tag{25}$$

通常利用模型的传输时间来间接地反映通信效率,即通过计算模型大小(比特)与网络数据速率(比特每秒)之间的比率来反映在数据传输过程中所需的时间。

$$\text{模型传输时间}=\frac{\text{模型大小(bit)}}{\text{网络数据速率(bit/s)}} \tag{26}$$

模型传输时间总和越长,代表客户端与服务器的通信开销越大,因此通信效率可以直接影响到遗忘的效率和全局模型的恢复速度。

遗忘效率指标可以评估算法的计算效率以及擦除贡献的效率。通过不同的遗忘效率指标对算法实验结果进行全面评估和分析,可以更严谨地证明方案的优越性。

3 联邦遗忘学习的隐私保护

联邦遗忘学习旨在通过忘记特定客户端或其数据贡献来实现数据隐私保护、保证模型安全,因此如何在遗忘过程中有效防范隐私泄露成为关键问题。本节从遗忘者的不同角度出发,对联邦遗忘学习中的隐私保护方案进行分析和介绍:

一类是服务器作为遗忘者,即服务器主导整个遗忘的流程,而客户端无需主动参与。服务器通过存储客户端上传的梯度更新或参数信息来执行遗忘操作,但这可能导致隐私泄露。为了解决此类隐私泄露问题,常用的方案是通过差分隐私技术对客户端上传的信息进行保护,从而保障联邦遗忘学习的隐私安全。

另一类是客户端作为遗忘者,遗忘过程主要在目标客户端上执行,而服务器则仅负责协调遗忘和聚合模型。然而,目标客户端执行遗忘操作后将会上传遗忘后的本地模型,那么恶意的服务器则可以通过分析遗忘前后的模型差异进行成员推理攻击,从而推断出参与训练的成员信息。为防止此类隐私泄露,通常通过集成防御相关攻击的方案到联邦遗忘学习中,从而确保隐私得到有效保护。

为了使对比分析更加清晰,我们总结了服务器和客户端分别作为遗忘者时,造成系统隐私泄露的原因以及易受攻击的方式。此外,我们根据可能受到的攻击类型,对比了差分隐私[58-59]与同态加密[62-63]两种隐私保护方法,分析了它们的计算开销及对模型精度的影响,如表 5 所示。

表 5 遗忘执行角色面临的隐私泄露威胁及隐私保护方法

遗忘执行者	造成隐私泄露原因	易受攻击方式	保护隐私方式	计算开销对比	影响精度对比
服务器	存储大量历史信息	模型反转攻击[56] 梯度泄漏攻击[57]	差分隐私方法[58-59]	计算开销较低,噪声产生轻微的影响	噪声的引入会不可避免地影响模型的精度
客户端	遗忘前后模型差异	成员推理攻击[60] 数据重构攻击[61]	同态加密方法[62-63]	计算开销较大,加密算法较复杂	不会直接影响模型的精度,因为在解密后的模型参数是准确的

3.1 服务器作为遗忘者

在接到遗忘请求后,服务器作为主导者,负责统筹整个遗忘流程。它通过收集和分析历史梯度及模型更新等信息,调整遗忘前的全局模型。在某些情况下,服务器可能会协调客户端,通过较少的重训练轮次获取更多的训练信息,而无需客户端主动参与。在联邦遗忘学习中,大多数方案通过利用历史信息来进行执行遗忘操作,以恢复全局模型,例如,文献[24]所提出的 FedEraser 方案,保留训练过程中客户端的历史参数更新,从而有效消除特定客户端数据对全局模型的影响,实现高效的数据删除。此外,文献[26]提出的 FUKD 方案通过减去模型中的历史累积更新来消除客户端贡献,并使用知识蒸馏方法恢复模型的性能。

需要注意的是,这种历史近似的恢复方式有可能会给联邦遗忘学习造成隐私风险,因为全局模型恢复的过程中,服务器不可避免地会存储一些客户端的历史信息。但服务器不一定总是诚实的,所谓半诚实服务器,是指虽然没有恶意,但对客户端数据感兴趣,并可能利用访问权限收集更多隐私信息,从而增加隐私泄露风险。更为严重的是,若服务器是恶意的,它可能通过利用这些历史信息发起模型反转攻击[56]和梯度泄漏攻击[57]。通过观察模型输出以及利用历史梯度信息,推断原始训练数据特征和重建训练数据样本,从而使得系统隐私泄露,如下图 3 所示。

差分隐私方法在保护客户端隐私数据方面起到了关键作用。通过在客户端训练模型时向梯度中注入随机噪声,从而确保服务器在接收和聚合梯度时无法准确获取每个客户端的数据。这种噪声的引入

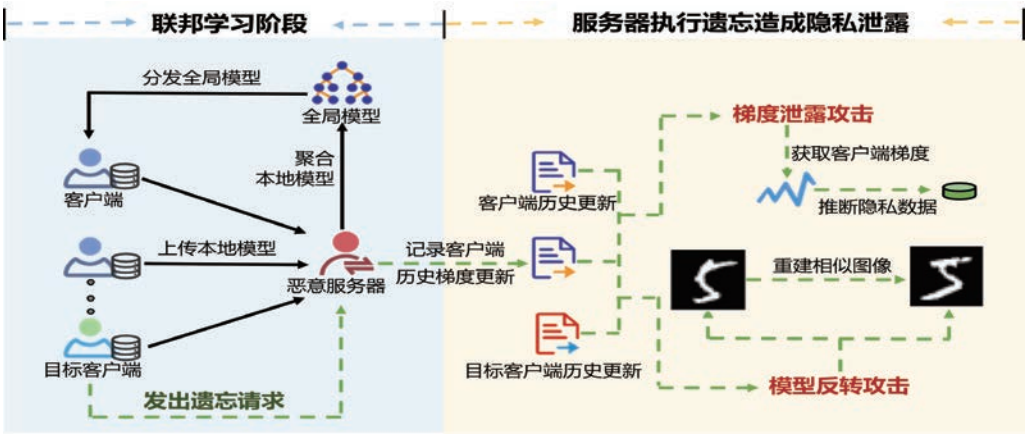

图 3　服务器作为遗忘者所引起的隐私泄露

使得即使服务器获取了梯度信息,也无法推断出具体的客户端数据。例如,Li 等人[58]提出的 SFU 方案中设计了一种差分隐私机制,旨在防止在联邦遗忘学习过程中因客户端表示矩阵传输而可能导致的隐私泄露。该方案通过向表示矩阵的每个向量添加随机扰动,不影响遗忘学习的同时,保证了隐私的保护。此外,Zhang 等人[34]提出了一种联邦遗忘学习框架 FedRecovery。FedRecovery 算法通过计算历史提交中的梯度残差来消除被遗忘客户端的影响,这样可以减少对重新训练的依赖,同时保证剩余客户端的贡献不受损害。特别的是,FedRecovery 通过在参数空间中引入高斯噪声来掩盖模型之间的差距从而实现差分隐私保护,这样既可以防止客户端的隐私被泄露,也可以确保被遗忘的模型与重新训练的模型在统计上难以区分。FedRecovery 方案不仅可以减少对计算资源的依赖和对剩余客户端的干扰,还可以为联邦遗忘学习框架提供稳健的隐私保障。因此,将其作为联邦遗忘学习中经典的隐私保护方案,如图 4 所示。

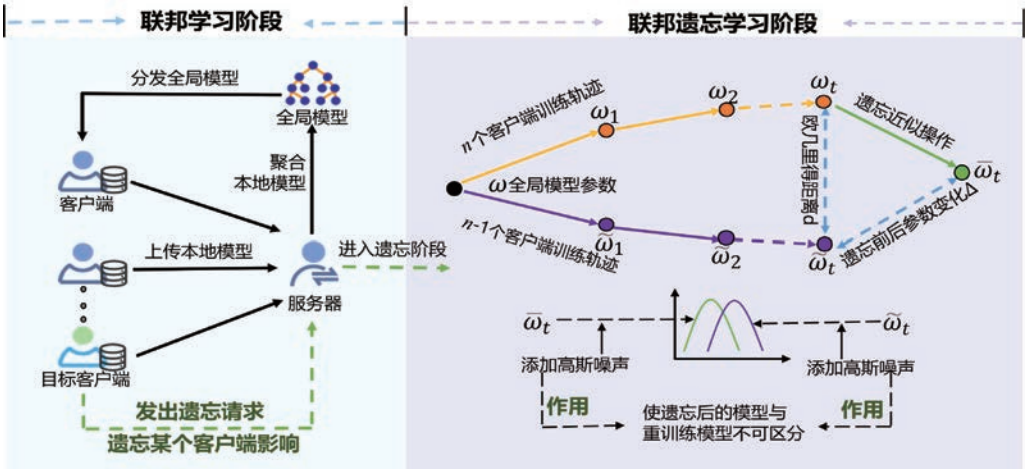

图 4　FedRecovery 算法隐私保护框架

在执行遗忘的过程中,除了可以使用差分隐私技术有效保护客户端上传的隐私信息外,还可以通过避免依赖客户端的历史信息来实现隐私保护。在 Halimi[44]等人所提出的方法中,客户端首先在本地执行遗忘学习,随后与服务器和其他客户端进行少量轮次的联邦学习,最终得到全局遗忘模型。该方法无需全局访问训练数据,也不要求服务器或客户端存储更新历史,从而有效保护了客户端的隐私,降低了数据泄露风险。

3.2　客户端作为遗忘者

当目标客户端作为遗忘者时,服务器只起到协调和聚合的作用,而执行遗忘的过程主要发生在目标客户端上。目标客户端可以主动地对自己本地的数据进行删除,然后参与到正常的训练过程中。联邦遗忘学习中的遗忘者可以是服务器、剩余客户端、所有客户端,或目标客户端。这些角色因功能和数据访问权限的差异,各有优劣。若关注隐私泄露问题,通常建议优先选择目标客户端作为遗忘者。这

是因为服务器参与时,需存储参数更新的历史记录,可能因此引发隐私泄露。而如果仅由目标客户端执行遗忘操作,则可随时根据需要进行,无需其他客户端付出额外的计算或通信成本[15]。

但是,仅仅删除目标遗忘数据可能无法完全清除其先前的影响,因此需要通过修剪和微调进一步提升遗忘效果。例如,Wang 等人[25]提出 FUCP 方法,旨在通过清除模型中特定类别信息,解决 CNN 分类模型的选择性遗忘问题。通过在目标客户端上进行通道修剪和微调以消除目标数据对其对模型的影响,从而确保无需对训练数据和存储的历史信息进行访问,有效降低了数据隐私泄露的风险。

虽然将目标客户端当作遗忘者可以避免服务器的梯度反转攻击和梯度重建,但是在成员推理攻击中,恶意服务器可以利用训练模型的随机梯度下降算法来发起白盒 MIA[60]。此时目标客户端是遗忘者,恶意服务器可以轻松地获得遗忘之前和遗忘之后的目标客户端的本地模型,并可以利用增强的 MIA 通过利用模型的中间计算进行攻击,即可以更有效地推断出目标客户端的训练数据信息。除此之外,遗忘前后模型的差异会包含遗忘信息,因此恶意服务器可以利用数据重构攻击近似地恢复出已经遗忘掉的目标数据信息[61],如图 5 所示。

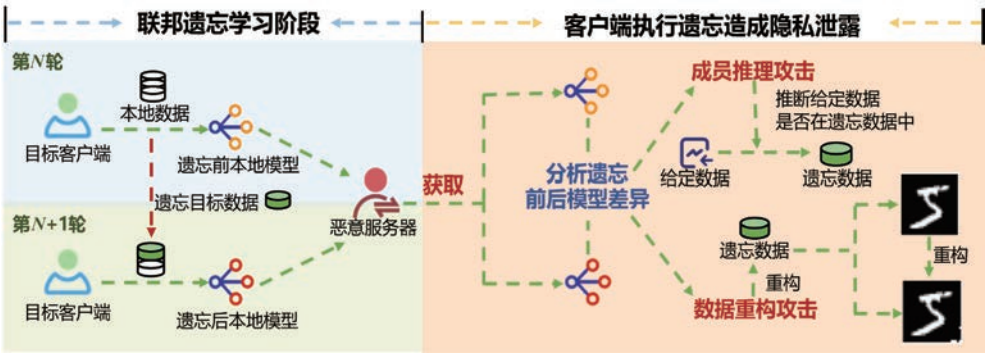

图 5 客户端作为遗忘者所引起的隐私泄露

在这种情况下,恶意服务器知道两个版本模型的架构和参数,从而导致置信度屏蔽方法[64]可能不再有效。因此,需要将有效的防御集成到联邦遗忘学习算法中,可以考虑为提高学习模型的泛化性而提出的 L_2 正则化技术[65],它通过降低目标模型过拟合程度的方法,从而很好地阻碍 MIA[66]。

同态加密作为一种先进的加密技术,能有效地抵御成员推理攻击和数据重构攻击,从而实现保护训练数据隐私的目的。同态加密技术允许在加密数据上直接计算,使攻击者无法直接访问原始数据,从而无法推断出输入数据是否在训练集中。此外,同态加密的操作在加密域中进行,减少了对明文数据的暴露,降低了数据重构攻击的风险。

尽管联邦遗忘学习的隐私保护研究尚显不足,但隐私问题至关重要。因此,将有效的防御机制融入联邦遗忘学习框架时,还应在保证模型鲁棒性和收敛性之间取得平衡。

4 联邦遗忘学习的模型恢复

在联邦遗忘学习中,全局模型的构建依赖于多个参与方的协同训练。因此,一旦遗忘过程导致模型性能下降,不仅会增加额外的优化成本,还可能影响联邦学习参与方对系统的信任和持续参与意愿。因此,联邦遗忘学习中的模型性能恢复对参与者和服务器双方都具有关键意义。模型性能恢复问题主要可以归纳为两类。

一类是目标数据未被彻底遗忘,仍残留于系统中。即使目标客户端或其数据在遗忘操作后被剔除,且不再参与后续训练,但其在此前训练轮次中对模型的影响可能已经传递给其他客户端。这些残留影响往往隐蔽,不易被发现,从而导致遗忘后的全局模型性能受到影响。

另一类是灾难性的联邦遗忘学习[67],当在遗忘过程中删除了一些关键的信息,例如代表性数据样本、稀有的类别样本或关键特征,遗忘后的模型与遗忘前的全局模型在性能上可能出现显著差异,导致模型的可用性和系统稳定性下降。

为了更清晰地总结联邦遗忘学习中影响模型性能恢复的因素及相应的遗忘方法,我们归纳了如下内容,如表 6 所示。

表 6　联邦遗忘学习中确保模型性能恢复方法总结

影响模型恢复类型	影响方式	确保模型性能恢复方法	关键技术
残留遗忘数据	未能完全清除目标数据先前对系统的影响	FUKD 中懒惰学习策略[26]	设置客户端在所有训练轮次中的更新值为 0,从而模拟它参与了训练
		RevFRF 中两级撤销协议[69]	第一级确保被撤销参与者数据无法被其他参与者使用;第二级确保被撤销的参与者无法继续参与训练
灾难性的遗忘	遗忘关键信息导致无法恢复模型性能	FedAF 中使用知识保留器[43]	知识保留器通过弹性权重巩固策略来约束模型中的关键权重变化
		REBFL 中记忆保留—动态惩罚策略[70]	记忆保留策略保留全局模型的记忆;动态惩罚策略是惩罚参与者过度遗忘

4.1　残留遗忘数据威胁模型性能恢复

联邦遗忘学习的一个重要目标就是通过删除目标数据的影响而确保模型不再依赖被遗忘数据,同时保持整体性能稳定。然而,由于训练过程具有交互性[41]和增量性[68],即便执行了遗忘操作,目标数据更新的影响仍可能残留在系统中,进而影响模型的性能恢复。

在联邦学习中,客户端的贡献是交互式的[41],模型聚合过程中,各客户端不断地将本地数据学习到的知识共享给全局模型。因此,即使某一客户端执行了遗忘操作,由于其他客户端仍可能间接保留相关信息,导致全局模型难以彻底消除遗忘数据的影响。此外,联邦学习的训练过程是增量的[68],每次局部更新都会继承客户端的历史更新信息。如果全局模型曾基于被遗忘数据进行调整,那么后续的模型更新仍可能隐含这些遗忘数据的影响,使得模型难以恢复到理想状态,如下图 6 所示。

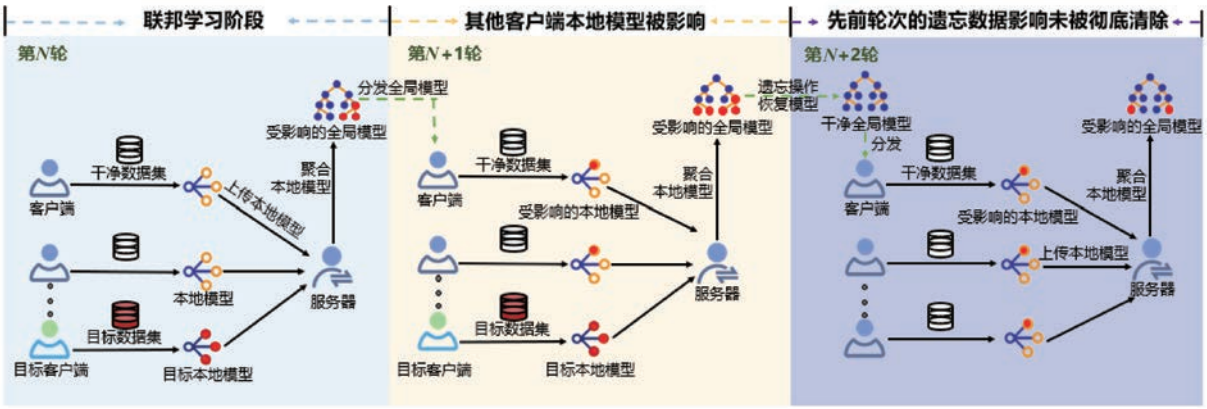

图 6　残留遗忘数据威胁模型性能恢复

在联邦遗忘学习的框架下,如何在保证遗忘有效性的同时,提高模型性能恢复能力是一个关键的问题。为此,RevFRF 方案[69]提出的采用了联邦叶子扩展协议和联邦最优查找协议,使参与者在不直接共享数据的情况下构建随机决策树(Random Decision Tree,RDT)。如图 7 所示,RevFRF 包括以下四个实体:服务器、一组参与者、计算服务提供者和密钥生成中心。服务器负责数据的加密和解密,以及模型的训练和预测;参与者则提供用于训练随机森林的数据;计算服务提供者协助服务器进行复杂的加密运算;密钥生成中心根据密钥管理的需求,负责去除过期密钥并向所有实体分发新的密钥。RevFRF 中的第一级撤销确保训练好的随机森林模型中诚实参与者的数据无法被其他参与者访问;第二级撤销则确保被遗忘的参与者无法继续利用其他参与者的模型数据。通过这两级撤销机制,系统能够有效验证残留的遗忘数据是否已被彻底清除,从而确保模型的性能恢复。

在应对残留遗忘数据威胁模型性能恢复时,文献[26]提出了另一种方法,利用知识蒸馏技术来消除客户端的贡献。通过减去模型中累积的历史更新,旨在恢复模型的准确性。然而,由于联邦学习的增量性,仅减去目标客户端的贡献可能导致显著偏差。因此,该文献进一步提出了一种懒惰学习策略,以更有效地消除目标客户端的影响,同时结合知识蒸馏方法恢复模型性能。值得注意的是,该方法在训练过程中引入了后门攻击,以评估遗忘的效果,证明其有效地消除了遗忘数据对全局模型的影响。

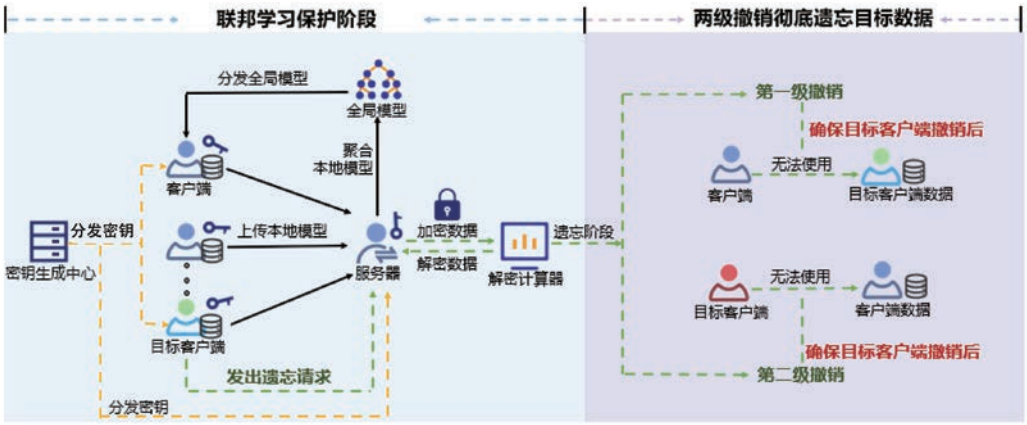

图 7 RevFRF 算法彻底遗忘目标数据框架图

4.2 灾难性联邦遗忘降低模型可用性

在联邦遗忘学习中，灾难性遗忘是指在执行遗忘操作时，模型虽然去除了目标数据的影响，但是导致整体性能大幅下降，影响未被遗忘数据的学习能力[67]。这一现象与持续学习(Continual Learning)中的灾难性遗忘存在一定联系，二者都涉及模型在面对数据变动时，旧知识的丢失以及模型性能的下降。然而，两者发生的机制和原因有所不同。在持续学习中，灾难性遗忘主要是由于模型在学习新任务时，旧任务的知识被新任务的训练所覆盖，导致模型在旧任务上的表现急剧下降，这是由于神经网络的梯度优化过程倾向于最小化当前任务的损失，而未对旧任务的知识进行有效的保护。而在联邦遗忘学习中，灾难性遗忘通常发生在数据分布异质的情况下，当遗忘特定数据或客户端的影响时，模型可能会失去关键任务相关的特征，导致其泛化能力下降，甚至使某些任务完全失效。由于联邦学习的数据是分布式存储的，遗忘后的数据难以恢复，这种不可逆的影响可能比持续学习中的灾难性遗忘更具挑战性。

为了更加清楚地展示灾难性遗忘对模型性能的影响，我们在 CIFAR-10 数据集上进行了实验评估，此数据集分为 10 个类别，每个类别有 6000 张图像，我们在实验中将类别为鸟的权重设置为 37% 来代表关键信息，而其他类别则设置为普通信息分别占比为 7%。在第 50 轮执行遗忘操作后，遗忘普通信息会导致模型的测试准确率小幅度下降，但遗忘关键信息则使得全局模型的测试准确率急剧下降，甚至可能导致模型无法使用，如图 8 所示。

为了防止灾难性联邦遗忘学习的问题，Alam 等人[70]在解决联邦学习中的后门攻击时提出了两种策略：记忆保留和动态惩罚策略。记忆保留策略

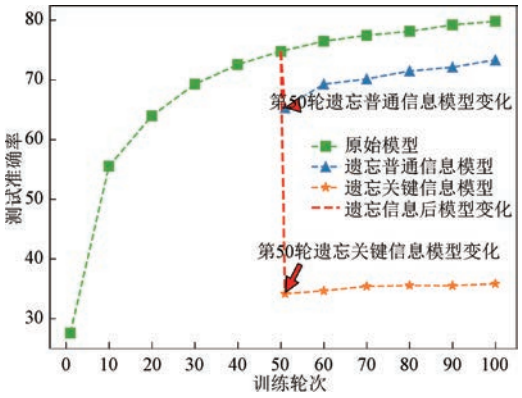

图 8 灾难性联邦遗忘学习实验示例

通过使用受损参与者的良好数据集保留全局模型的记忆；而动态惩罚策略则实施一种动态惩罚机制，以惩罚与触发数据集和触发模式无关的过度遗忘。这两种策略不仅有助于在去除后门攻击时有效保持模型性能，还减少了服务器端潜在的安全风险。

然而，记忆保留策略需要额外的存储和计算资源来维护和处理受损参与者的良好数据集，这会给系统带来较大的计算负担。因此，Li 等人[43]借鉴神经学中的主动遗忘过程，利用新的记忆覆盖旧的记忆，从而提高遗忘过程的效率。该框架主要通过记忆生成器生成无效知识且易于学习的新记忆以覆盖旧记忆，利用知识保留器通过可塑权重巩固策略弹性地约束权重变化，以避免灾难性遗忘现象。这种策略具备更高的灵活性，能在不影响整体模型性能的前提下保留更多有用的知识。

联邦遗忘学习中的模型性能恢复问题应引起重视。如果遗忘过程导致模型性能下降，整个系统的有效性和实用性可能会受到影响。因此，在设计遗忘算法时，需权衡遗忘效果与模型性能恢复的平衡，确保系统在满足遗忘需求的同时，仍能保持良好的

预测能力和稳定性。

5　联邦遗忘学习的计算效率

在大规模的联邦遗忘学习环境中,模型训练和更新可能涉及大量参与方和海量数据。提高计算效率可以显著减少训练和更新模型所需的时间,从而加快模型的部署和应用速度。从计算效率的角度来看,如何高效地进行"遗忘"并且恢复出准确的全局模型无疑是联邦遗忘学习中的一大难点[71]。

在联邦遗忘学习过程中,直接移除特定客户端并从头开始训练虽然方法直观,但面临双重现实约束:其一,从头开始训练带来的计算成本呈指数级增长(复杂度达 $O(n^2)$),难以应对高频次遗忘请求;其二,由于联邦学习的持续聚合特性,待移除客户端的历史参数更新已通过多轮梯度交换(如 FedAvg 算法)渗透至其他客户端,导致贡献追溯困难。现有的方案主要通过利用历史信息遗忘和快速重新训练来实现联邦遗忘学习,在这两种遗忘方式的背景下,研究者设计了各自的优化算法以提升计算效率。

5.1　基于历史信息遗忘的计算效率

在联邦学习中,客户端根据本地数据样本训练模型,服务器仅聚合客户端本地模型更新。这种不允许共享原始数据的机制为遗忘操作带来了挑战。要从训练好的全局模型中删除数据样本,一种直接的方法是按照原始训练过程,在删除要遗忘的数据后从头开始训练新模型。然而,这在现实中显然不可行。因此,研究者们为了提升遗忘效率而提出了多种基于历史信息进行遗忘的算法,我们将根据遗忘的执行者将其分为中央服务器执行遗忘和目标客户端执行遗忘两类进行综述。

5.1.1　中央服务器执行遗忘

中央服务器拥有所有客户端的历史模型更新纪录和模型参数,因此可以直接从全局模型中删除特定客户端的影响,使得遗忘过程统一且可控。FedEraser 方法[24]通过校准保留的客户端历史更新,使中央服务器能够快速构建遗忘后的模型。校准如公式(27)所示:

$$\widetilde{U}_k^l = \frac{\| U_k^l \|}{\| \| U_k^l \| \|} \cdot \widetilde{U}_k^l \tag{27}$$

其中,$\widetilde{U}_k^l$ 代表第 k 个客户端在第 l 轮训练中的更新,$\widetilde{U}_k^l$ 是归一化后的更新,表示了如何调整模型参数。这样 FedEraser 能够有效消除了目标客户端数据对全局模型的影响,并显著减少了构建遗忘学习模型的时间,与从头开始重新训练相比,速度提高了 4 倍。

然而,FedEraser 方法不仅依赖于其他客户端的参与,还需要在客户端和服务器之间进行额外的通信。为了避免其他客户端的参与所引起的计算和通信开销的增加,Wu 等人[26]提出了 FUKD 方法,其主要通过从模型中减去累积的历史更新来消除目标客户端的贡献,然后利用在中央服务器进行的知识蒸馏方法恢复模型的性能。当目标客户端作为攻击者时[31],可以从全局模型中减去其所有历史平均更新,以完全删除攻击者对模型的影响。接着,通过知识蒸馏修复因减去攻击者历史参数更新而造成的偏差,实现有效遗忘。具体而言,FUKD 可以通过以下公式来表示:

$$M'_F = M_1 + \frac{N}{N-1}\sum_{t=1}^{F-1}\Delta M_t - \frac{1}{N-1}\sum_{t=1}^{F-1}\Delta M_t^N + \sum_{t=1}^{F-1}\epsilon_t \tag{28}$$

其中,M'_F 代表遗忘后的全局模型,它是对初始全局模型 M_1 进行调整后得到的模型。$\frac{N}{N-1}\sum_{t=1}^{F-1}\Delta M_t$ 表示对前 $F-1$ 轮的客户端更新量进行加权平均,以调整全局模型。$-\frac{1}{N-1}\sum_{t=1}^{F-1}\Delta M_t^N$ 则表示从全局模型中减去目标客户端的贡献,消除其对模型的影响。最后,修正项 $\sum_{t=1}^{F-1}\epsilon_t$ 用来进一步调整全局模型,消除因去除目标客户端数据所带来的偏差。

知识蒸馏依赖于教师模型(遗忘前的模型)和学生模型(遗忘后的模型)之间的知识传递。如果教师模型性能较差,学生模型的性能可能难以达到预期,导致模型精度下降。因此,Zhao 等人[36]提出了一种基于动量衰减(Momentum Degradation,MoDe)和记忆引导的知识消除策略,中央服务器在接收到目标客户端发出的遗忘请求之后,采用 MoDe 策略消除模型中隐式知识。为了减轻消除带来的性能下降,记忆引导策略对不同的数据点进行微调,有效恢复模型在剩余数据点上的区分能力。该策略解耦了联邦遗忘过程与训练过程,使得遗忘过程可以应用于任何联邦学习模型架构,并且与从头开始训练相比,执行时间加速了 5 至 20 倍。

尽管记忆引导策略有效,但在对数据点进行微调时,可能会产生额外的计算和时间开销,尤其是在

处理大规模数据集或复杂模型时。相对而言，Zhang 等人[34]提出的 FedRecovery 高效联邦遗忘算法无需重新训练或微调修复模型偏差，而是通过引入梯度残差量化被遗忘客户的影响，并从全局模型中去除其加权和，权重根据客户对降低全局损失的贡献评估得出。该算法仅在中央服务器运行，无需与其他客户端进行额外的交互，从而显著减少了计算与通信开销。

中央服务器不仅可以处理历史信息，还能进行模型预训练，以提升模型的初始性能并加速后续训练。在 Jin 等人[32]提出的线性联邦遗忘学习框架 2F2L 中，中央服务器首先使用自身的数据进行模型预训练。随后基于神经切迹核（Neural Tangent Kernel，NTK）[72]的启发，优化训练模型的二次损失函数。在移除阶段，2F2L 中的 FedRemoval 方法在不访问其他剩余客户端隐私数据的情况下，能够有效移除模型中的目标数据，同时保持模型准确性，避免了重新训练的时间开销。

在对目标数据或者目标客户端进行遗忘之后，许多方法缺乏验证机制，难以确保数据在遗忘后不再对模型产生影响，也无法系统地评估遗忘效果的完整性和准确性。因此，Gao 等人[27]提出了一个整合联邦遗忘和验证的统一框架 VERIFI，通过集成遗忘和验证，系统地分析遗忘的效果及其影响。目标参与者被赋予验证权，能够在离开前通知服务器并在接下来的几轮通信中验证遗忘效果。然而，在大规模联邦学习环境下，与服务器进行多轮次的交互验证会增加系统中的通信负担。Xiong 等人[38]提出的 Exact-Fun 方法则避免了对客户端的依赖。当目标客户端提交遗忘请求时，中央服务器会处理请求并计算新的本地模型，上传后聚合生成新的全局模型。Exact-Fun 采用量化评估机制，通过比较模型在遗忘前后的输出分布自动评估遗忘效果，若分布相同，则可以直接结束遗忘过程，减少不必要的计算。

在大规模的多客户端联邦学习系统中，随着隐私需求的变化，客户端可能频繁提出数据删除或遗忘的请求。这些连续的遗忘请求对系统提出了更高的要求：既要确保模型在每次数据移除后仍能保持良好的性能，又要尽量减少时间开销。为此，Fraboni 等人[35]提出了 IFU 方案，旨在从联邦学习模型中移除特定客户端的贡献。在 IFU 的基础上进行扩展得到的 SIFU 方案可以用于处理多个目标客户端连续发出的遗忘请求。SIFU 无需从头开始重新训练模型，而是从合适的中间模型出发，开展遗忘过程。此外，该方案还通过引入随机机制对选定的中间模型进行扰动，在保证模型去噪的同时，尽量减少随机梯度下降的步数，以实现高效的联邦遗忘学习。

中央服务器利用客户端在先前轮次中上传的历史信息来执行遗忘，从而更好地主导整个遗忘过程，避免过多增加其他客户端的计算成本和通信开销。该方法不仅关注高效删除目标数据，还解决了在数据删除后如何验证遗忘效果的问题，并有效处理了多次连续的遗忘请求。然而，在执行遗忘时，服务器对全局模型的控制权可能过大，从而增加隐私泄露的风险。因此，在实现遗忘的过程中，如何平衡中央服务器的控制与客户端的隐私保护，仍然是一个亟待解决的重要问题。

5.1.2 目标客户端执行遗忘

通过删除目标客户端上的目标数据，并对模型进行微调或校准，可以有效消除该数据对模型的影响，同时保持其他数据的完整性。这种方式减少了服务器访问频率，降低了敏感数据泄露风险，从而提升了隐私保护效果。Wang 等人[25]提出的 FUCP 方案则不需要全局访问训练数据，而是在目标客户端上通过通道修剪和微调来实现遗忘。他们指出 FedEraser 方案所进行的客户端级遗忘学习在卷积架构中只有浅层网络才有可能进行，所以提出了 FUCP 方案用于解决 CNN 分类模型中的选择性遗忘问题，通过引入词频－逆文档频率（Term Frequency-Inverse Document Frequency，TF-IDF）来量化信道的类别区分，并修剪目标类别中最相关的信道，以消除其对模型的贡献，从而确保在不降低模型精度的情况下提升遗忘的速度。

然而，在处理高度复杂的非线性数据模式时，修剪相关通道可能无法完全消除类别影响。此外，若 TF-IDF 未能准确量化信道的类别区分，可能导致误删关键信道，从而削弱模型性能。因此，Che 等人[33]提出了一种基于非线性泛函理论的联邦遗忘学习方法 FFMU，采用随机梯度平滑和量化方法（Prompt Certified Machine Unlearning，PCMU）[73]，对每个目标客户端上的本地模型实现快速遗忘。PCMU 方法通过同时进行训练和遗忘操作来提高效率，在每个目标客户端上生成一个本地遗忘模型，利用非线性函数分析理论将本地遗忘模型重新定义为 Nemytskii 算子[74]的输出函数。理论分析表明，Nemytskii 算子光滑可微，从而生成光滑流形。通过这些方法，FFMU 成功地提高了联邦遗忘学习算

法的效率。虽然这种方式可以避免删除不必要的数据,但是生成本地遗忘模型所需的计算成本急剧增加。然而,由 Wu 等人[28]提出的基于逆随机梯度上升的联邦遗忘学习框架(Stochastic Gradient Ascent-Elastic Weight Consolidation,SGA-EWC)则可以避免因需要计算本地遗忘模型而引起的成本增加,其主要是通过反转梯度来消除特定训练数据对模型性能的影响。EWC 是一种正则化技术,通过限制参数更新来保留先前任务的重要信息。其实现方式是计算每个参数的重要性因子,并将其作为正则项添加到交叉熵损失中[75]。该框架针对三种不同类型的请求(即样本、类别、客户端)进行了优化,以实现高效的联邦遗忘学习。

无论是 FFMU 方案中所需计算的本地遗忘模型,还是 SGA-EWC 框架中所需计算的重要性因子,这些过程都会不同程度地产生计算和存储开销。为解决这一问题,Yuan 等人[29]提出了 FRU(Federated Recommendation Unlearning)方法,通过修改联邦推荐系统 FedRec[76]的历史更新,在目标客户端上进行回滚和校准历史参数更新以实现遗忘,利用这些更新来加速 FedRec 重建,从而提升遗忘效率。然而,FRU 在复杂数据集上进行遗忘是具有挑战的。随着数据集规模和模型复杂性的增加,遗忘方法难以兼顾计算效率与效果。因此,Dhasade 等人[37]提出的 QUICKDROP 方案通过利用数据集蒸馏(Dataset Distillation,DD)技术,使得在训练完成后产生一个与原始训练数据集相当的合成数据集,从而显著降低遗忘学习和恢复阶段的计算开销。具体地,蒸馏数据集 S 是通过以下公式生成的:

$$\theta_S=\arg\min_{\theta}L_S(\theta)=\frac{1}{|S|}\sum_{(s,y)\in S}\ell(\phi_\theta(s),y) \tag{29}$$

其中,$\ell(\cdot,\cdot)$ 是任务特定的损失函数,θ_S 是通过蒸馏数据集最小化损失函数 L_S 的参数。然而,将 DD 与联邦学习集成时,蒸馏数据集的质量可能略有下降,导致恢复阶段模型性能降低。为提高蒸馏数据集质量,QUICKDROP 允许客户端进行额外的微调步骤。具体来说,客户端会利用以下公式进行更新:

$$\theta_{t+1}=\theta_t-\eta\nabla_\theta L(\theta_t) \tag{30}$$

其中,η 是学习率,$L(\theta_t)$ 是当前参数 θ_t 对应的损失。虽然这会增加额外的计算成本,但整体训练时间比从头训练减少了 463.8 倍。此外,QUICKDROP 方案还可以处理多个连续的遗忘请求,通过以下更新公式避免了重新训练模型所需的大量计算开销:

$$\theta_{k+1}=\frac{1}{N}\sum_{i=1}^{N}\theta_i \tag{31}$$

其中,θ_{k+1} 是聚合后的全局模型,θ_i 是每个客户端的本地模型参数。

在实际的联邦遗忘学习场景中,不仅需要考虑数据集和模型的复杂性,还必须应对数据的异构性和分布不均衡问题。这种异构性意味着各个客户端可能拥有不同的数据特征和规模,从而影响遗忘的精度与效率。因此,设计一种在异构环境下确保遗忘效果的联邦遗忘学习机制显得尤为关键。Zhu 等人[30]提出的 FedLU 是一种异构知识图谱嵌入的联邦学习和联邦遗忘框架。在联邦学习中使用相互知识蒸馏方法来平滑地在客户端之间交换知识,实现全局收敛和局部优化。该方法有助于提高异构数据下的预测精度。同时,在联邦遗忘学习中,结合认知神经科学中的回顾性干扰和被动衰减方法,实现了三元组的遗忘和传播。通过删除局部目标客户端的特定知识,并利用知识蒸馏将其传播到全局模型,实现高效的联邦遗忘学习。

目标客户端执行遗忘能有效消除目标数据的影响,并且可以减少服务器访问频率,降低隐私泄露的风险。这类方法关注如何遗忘目标数据,同时着重提高遗忘效率和保持模型性能,尤其是在复杂和非线性数据模式下。此外,目标客户端执行遗忘在处理多次遗忘请求和适应异构数据环境方面也表现良好。然而,计算成本和存储开销可能会随着模型复杂性增加而上升。因此,未来研究需优化遗忘效率与计算资源的平衡,以确保隐私保护与模型性能的双重需求。

5.2　基于快速重新训练的计算效率

快速重新训练通过优化重训练算法或局部重训练,旨在提升完全遗忘目标数据影响时的计算效率。基于历史信息遗忘虽然可以通过近似恢复全局模型的方式达到快速遗忘的效果,但是其在响应 GDPR 等监管政策方面可能有所欠缺,因为 GDPR 要求彻底消除目标数据的影响。

基于历史信息的遗忘方案,例如 FedEraser,通过服务器利用客户端缓存的历史梯度来近似遗忘后的梯度,但这一方法无法完全消除数据样本对训练模型的影响。因此,Liu 等人[41]提出了一种名为 Rapid Retraining 的方法,如图 9 所示,它借助一阶泰勒展开技术,旨在实现对目标数据样本的完全擦除。该算法通过以下公式有效地利用梯度和曲率信息来寻找更优的下降方向,显著减少了重训练所需

的时间：

$$\nabla F_k(\omega^*)+\boldsymbol{H}_k(\omega^*)(\omega_u-\omega^*)\approx 0 \quad (32)$$

其中，$\boldsymbol{H}_k(\omega^*)$ 是海森矩阵，ω_u 是经过“遗忘”数据后的新模型。为了进一步提升效率，该方法通过引入对角经验费舍尔信息矩阵(FIM)来近似海森矩阵的逆，得到更新规则：

$$\omega_{t+1}=\omega_t-\frac{1}{B-\Delta B}\boldsymbol{\Gamma}^{-1} \quad (33)$$

其中，$\boldsymbol{\Gamma}$ 是 FIM 的对角矩阵，Δ 是当前的梯度。通过引入自适应动量技术，该方法能够降低近似误差，从而在计算速度和精度保持方面表现出较高的优势。

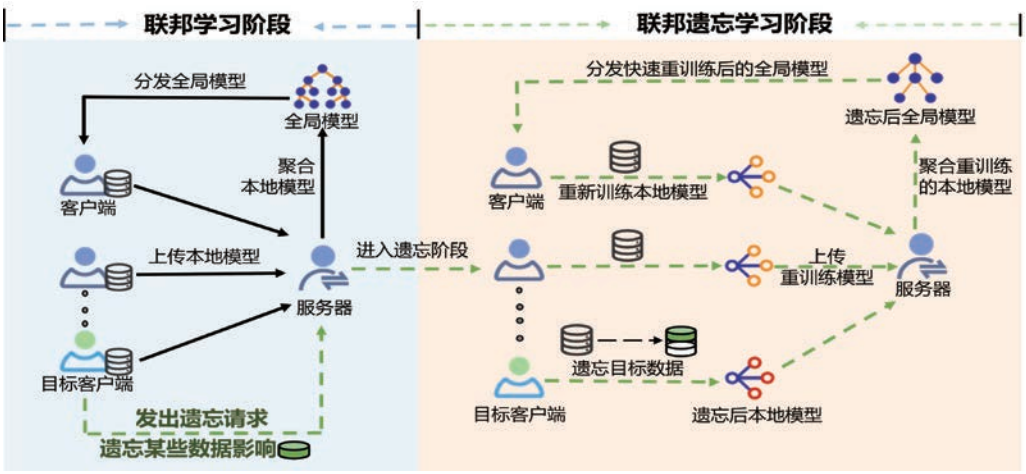

图 9 Rapid Retraining 算法提升计算效率框架

然而，服务器在利用历史信息执行遗忘时，不仅可能导致目标数据残留，还存在泄露隐私的风险。例如，客户端退出训练后，服务器仍可能使用这些参数进行进一步训练。为了解决这一问题，Lin 等人[40]提出了一种基于 Chameleon 哈希函数的可信联邦遗忘学习框架。通过验证模型更新的正确性和数据所有权，确保目标客户的数据效果被完全移除。具体来说，服务器在接收到遗忘请求后，使用以下公式对全局模型进行更新：

$$U_{tu+1}=\frac{\sum_{k=1,k\neq ku}^{K} w_k U_{tu_k}}{K-1} \quad (34)$$

其中，w_k 是每个客户端的权重，U_{tu_k} 是第 k 个客户端的模型更新，K 是客户端总数。然后，服务器根据以下公式进行全局校准：

$$M_{tu+1}=M_{tu}+U_{tu+1} \quad (35)$$

其中，M_{tu} 是当前的全局模型，U_{tu+1} 是新的局部模型更新。此外，框架中还开发了一种自适应贡献的重训练机制，利用 Gompertz 函数[77]评估目标客户在每轮训练中的贡献，从而有效确定最优遗忘轮数，保持模型准确性的同时显著降低计算开销。具体的贡献度评估公式为

$$\theta_{tk}=\arccos\left(\frac{\langle\nabla F(w(t)),\nabla F_k(w(t))\rangle}{||\nabla F(w(t))||\,||\nabla F_k(w(t))||}\right) \quad (36)$$

其中，$\nabla F(w(t))$ 是全局梯度，$\nabla F_k(w(t))$ 是第 k 个客户端的本地梯度。

FUCP[25]和 SGA-EWC[28]等方法主要是通过剪枝网络层和优化损失函数来进行近似实现目标数据的删除，但这些近似操作未能彻底清除目标数据的影响。针对这一问题，Lin 等人[39]设计了一种可扩展的联邦遗忘学习框架 SE，该框架将联邦学习和遗忘过程分为多个阶段，在每个阶段构建多个隔离分片并进行编码计算，通过在各分片中重新训练，以高效去除目标客户端的数据影响，同时保持模型的准确性。这种方法不仅提高了计算效率，也确保了模型性能。

尽管 SE 框架通过隔离分片降低了计算复杂度，但仍需进行多阶段的重训练，且未考虑异步并行处理，因此其并不适用于大规模的异构数据场景。为此，Su 等人[42]提出的 KNOT 算法通过聚类优化和异步执行，实现了更高效的大规模联邦遗忘学习。在 KNOT 中，客户端被分配到合适的聚类，聚合仅在每个聚类的内部进行。在执行遗忘操作时，仅在目标客户端所属的聚类内进行重训练，而无需所有客户端参与。这不仅显著提升了计算效率，还使系统在处理数据时更加灵活。

此外，主动遗忘在神经科学中被视为优化大脑记忆的积极过程。与传统观点“遗忘是被动的记忆衰退”不同，主动遗忘被认为是一种有效机制，旨在

移除无用信息以优化认知功能。基于此,Li 等人[43]提出的 FedAF 框架借鉴了神经学中主动遗忘的理念,通过新的记忆覆盖旧的记忆。该框架包括记忆生成器和知识保留器两个模块。记忆生成器生成易于学习的新记忆以覆盖无效知识,而知识保留器则通过可塑权重策略约束权重变化,帮助保留非目标数据的影响。这一设计不仅提升了计算效率,还增强了模型的整体效用。

无论是在 SE 框架中的隔离分片算法还是 KNOT 中的聚类算法,都可能对其他客户端造成不同程度的影响,从而影响全局模型性能。为此,Halimi 等人[44]提出了一种新的框架,通过反转学习过程实现局部遗忘学习,有效去除目标客户端数据对全局模型的影响。该方法限制模型参数在 l_2 范数球内,以其他客户端的局部模型为参考,保留其所学知识,同时使用投影梯度下降进行高效的局部遗忘学习。这种方法在实现接近重新训练的精度时,所需训练轮次显著减少,极大提升了计算效率。

在联邦遗忘学习领域,如何有效地实现对目标数据样本的遗忘,同时保障模型的准确性,已成为当前计算效率提升的重要研究方向。不同的执行方式各具特点:中央服务器执行遗忘具备全局统一和可控性优势,但在计算开销和隐私风险上存在隐患;而目标客户端执行遗忘则能在较高异构性和非线性复杂的场景中更好地保障隐私,尽管其计算和存储成本会随模型复杂度增加而上升。此外,快速重新训练重在响应高效移除目标数据影响的需求,尤为适合遵循 GDPR 等隐私法规的应用场景。

总体而言,联邦遗忘学习中计算效率的核心挑战在于如何进一步提升遗忘与重新训练的计算效率、优化遗忘效果的验证机制,并快速响应参与者在联邦学习中频繁地发起遗忘的请求。未来研究需在兼顾计算效率、隐私保护和模型性能的前提下,深入探索实现资源开销与计算效果的最佳平衡,以支持在更大规模和更高异构性的数据分布环境下的应用需求,从而提高系统的适应性和灵活性。

我们将所总结的联邦遗忘学习算法与从头开始训练和经典算法对比列为如表 7 所示,同时分析并总结了每个算法的时间复杂度。为了使对比更加可靠,我们使用更加细致的复杂度表示,而非简单的 $O(n)$。我们定义通用的符号如下:P 表示算法所用模型的参数数量(数量越多表示模型越复杂),原始数据集大小定义为 N,L 为全局训练轮数。

6　联邦遗忘学习的存储效率

在现实世界的联邦遗忘学习中,参与方通常分布在不同地理位置的终端设备上,这些设备的存储资源有限,无法存储大量的历史信息或梯度更新,因此节省存储空间对于资源受限的环境(如移动设备)至关重要。当需要进行数据擦除或删除某一客户端的全部贡献时,大多数现有研究方法依赖于存储之前训练轮次的历史信息,包括历史梯度和模型更新。这种方法虽然有效,但会显著增加存储成本,不仅在资源受限环境中部署和运行模型更困难,还可能影响模型效用和响应时间。此外,过高的存储成本还可能限制用户的扩展能力,阻碍其加入联邦学习系统,从而影响系统的规模和覆盖范围[78]。因此,提高联邦遗忘学习的存储效率具有重要意义。

利用客户端的历史信息可以提升存储效率,但这可能导致隐私泄露。相对而言,不依赖历史信息虽然能有效规避隐私风险,却可能增加计算开销。因此,我们可以根据方法是否依赖历史信息将提升存储效率的算法分为两类。

6.1　依赖历史信息提升存储效率

在联邦遗忘学习中,利用历史信息的方法通过记录和利用模型的过往参数、更新及其他相关信息,能够有效提升存储效率和计算性能。这类方法通常需要维护一定量的历史记录,以加速遗忘或回滚操作中的模型更新。例如,FedEraser 方案引入了一种新的存储和校准技术,针对参数更新中的信息前向耦合问题进行了解决。在全局模型训练过程中,中央服务器定期保留客户端的更新信息和对应轮次的索引,从而进一步校准这些更新以重构全局模型。这不仅有效减少了目标客户端数据对模型的影响,也提升了存储效率。

然而,FedEraser 方案在存储效率方面仍有改进空间,因为它过于依赖其他客户端的参与,需额外存储其模型参数。因此,基于知识蒸馏的方法,如 FUKD[26],通过减去模型中的历史参数更新来消除客户端数据,利用知识蒸馏恢复模型性能,避免了额外的存储需求,从而在存储效率上具有优势。

此外,FedEraser 方案不仅需要客户端的重新参与,还需存储大量模型检查点,这会导致较高的存储开销。为此,FedRecovery 算法[34]通过梯度残差量化被遗忘客户端对全局模型的影响,移除加权梯度残差,从而避免对模型的微调或校准需求,降低了

表 7 联邦遗忘学习算法的加速比

算法/框架名称	对比 Retrained from Scratch	对比 FedEraser[24]	时间复杂度	其他
FUCP[25]	CIFAR-10 上提升 8.9-11.01 倍 CIFAR-100 上提升 8.4-9.9 倍	CIFAR-10 上提升 2.2-4.8 倍	$O(L\times N\times P)$	优于基于 Fisher 信息遗忘方法
FUKD[26]	MNIST 上提升 6.5 倍 CIFAR-10 上提升 11.9 倍 GTSRB 上提升 29.3 倍	CIFAR-10 上提升 3.3-8.2 倍	$O(k\times\lvert X\rvert\times P)$ k 为蒸馏轮数，$\lvert X\rvert$ 为蒸馏时所用无标签数据集大小	蒸馏后的遗忘学习模型测试精度与从头开始重新训练的模型的测试精度几乎相同(差异小于 1%)
VERIFI[27]	CIFAR-10 上提升 377 倍	CIFAR-10 上提升 70.1 倍	$O(T\times N\times P)$ T 为微调轮数	比 GGS[47]在 CIFAR-10 上提升 8,6 倍 比 IGS[12]在 CIFAR-10 上提升 1.8 倍
SGA-EC[28]	MNIST 上提升约 50 倍		$O(N\times P)$	适用于不同级别遗忘请求
FRU[29]	MovieLens-100k 上提升 7 倍 Steam-200k 提升 7 倍以上		$O(L\times N\times P)$	在恢复被攻击的联邦推荐系统方面表现出了更好的性能和效率
FedLU[30]	FB15k-237-C3 上提升约 3 倍 FB15k-237-C5 上提升约 4-5 倍		$O(L\times N\times P^2)$	还提出了一种基于认知神经科学的遗忘方法
UBA[31]	MNIST 上提升 6.3 倍 CIFAR-10 上提升 11.5 倍 GTSRB 上提升 30.3 倍		$O(L\times N\times P)$	减去攻击者的历史参数更新 来消除全局模型上的后门
FFMU[33]	CIFAR-10 上提升 1.9 倍 到 3.4 倍	CIFAR-10 上提升 4.6 倍到 7.4 倍	$O(L\times N\times P)$	比 Rapid Retraining[41] 在 CIFAR-10 上提升 1.5 倍到 2.8 倍
FedRecovery[34]	MNIST 上提升约 679 倍 CIFAR-10 上提升约 1242 倍 SVHN 上提升约 1380 倍 USPS 上提升约 285 倍	MNIST 上提升约 385 倍 CIFAR-10 提升约 1082 倍 SVHN 上提升约 622 倍 USPS 上提升约 137 倍	$O(N\times P)$	效率大幅度提升，但是恢复的全局模型准确率降低，在 MNIST 上的恢复准确率只有 90.9%
SIFU[35]	MNIST、FashionMNIST、 CIFAR-10、CIFAR-100、 CelebA 上提升约 1 倍		$O\left(\sum_{r=1}^{R}(L\times r\times N\times P)\right)$ R 为总共需要处理的遗忘请求次数	IFU 通过计算每个客户端对全局模型的贡献，从而确定在 FL 过程中进行遗忘操作的最佳位置
MoDe[36]	MNIST 上提升 8.4～12 倍 F-MNIST 上提升 6.3～15.0 倍 CIFAR-10 上提升 4.2～7.6 倍 CIFAR-100 上提升 8.2～10.3 倍		$O(L\times N\times P)$	比 FUKD[26]在 MNIST 上提升 12 倍 比 UPGA[44]在 MNIST 上提升 2.4-5.25 倍 比 FUCP[25]在 CIFAR-10 提升约 1.67 倍
QUICKDROP[37]	CIFAR-10 上提升 463.8 倍		$O(L\times E\times N\times P)$ E 为客户端本地更新轮次	相比于 SGA-EWC[28] 提升 67 倍、相比于 FUCP [25]提升 65.1 倍
Exact-Fun[38]	特定条件下，在 Fashion-MNIST 上可提升超 10000 倍		最好情况 $O(L\times P)$ 最坏情况 $O(L\times N\times P)$	适当的量化参数 α 可以帮助 Exact-Fun[38] 比 Rapid Retraining[41]效果更好
SE[39]	MNIST 上提升 5.85 倍 FMNIST 上提升 5.77 倍 CIFAR-10 上提升 5.46 倍 Shakespare 上提升 16.3 倍	MNIST 上提升 3.04 倍 FMNIST 上提升 3.01 倍 CIFAR-10 上提升 2.87 倍 Shakespare 上提升 8.08 倍	$O(S\times((T+N)\times P))$ S 为分片数量 T 为重训练轮数	相比 Rapid Retraining[41] MNIST 上提升 4.05 倍、FMNIST 提升 3.90 倍、CIFAR-10 提升 3.67 倍
BCETFU[40]	MNIST、FMNIST、 CIFAR-10 上提升 1 倍以上		$O(L\times N\times P)$	与 Rapid Retraining[41]相比几乎有相同性能，但是训练时间却较少
Rapid Retraining[41]	CIFAR-10 上提升 7.1 倍 CelebA 上提升 9.6 倍		$O((L+E)\times N\times P)$ E 为客户端本地快速重训练轮次	
KNOT[42]	CIFAR-10 上提升 73% FEMNIST 上提升 45% Purchase-100 提升 61%	CIFAR-10 上提升 85% FEMNIST 上提升 47% Purchase-100 提升 56%	$O(N\times P\times M+M^3)$ M 为聚类的数量	比 Rapid Retraining[41]减少约 200%内存占用，并显著提高了收敛速度

续表

算法/框架名称	对比 Retrained from Scratch	对比 FedEraser[24]	时间复杂度	其他
FedAF[43]	MNIST 上提升 15.2 倍 CIFAR-10 上提升 95.9 倍 CelebA 上提升 31 倍		$O((Q\times N+E)\times P)$ Q 为教师模型数量， E 为在新记忆上训练轮数	由于牛顿优化器所需的精确计算，Rapid Retraining[41] 花费时间比 Retrained from Scratch 多
UPGA[44]	MNIST 上提升 2.7 倍 EMNIST 上提升约 2.9 倍 CIFAR-10 上提升约 5.5 倍		$O((L+E)\times N\times P)$ E 为客户端本地训练轮次	不需要服务器或其他客户端记录历史更新的参数

注：P 表示算法所用模型的参数数量，N 表示原始数据集大小，L 表示全局训练的轮数。

存储开销。同时，该算法采用早期停止策略，根据模型在验证集上的性能监控，及时终止训练，以进一步提高存储效率。

在提升算法的存储效率方面，除了减少客户端历史信息的存储外，蒸馏大型数据集也是一个有效的策略。Dhasade 等人[37]提出的 QUICKDROP 方法利用数据集蒸馏技术将原始训练数据集中的关键信息精简到更小的数据集中。该方法将数据集蒸馏与联邦学习训练过程结合，使客户端能重新使用训练过程中计算的梯度更新进行数据集蒸馏，并利用历史梯度信息匹配模型在原始和蒸馏数据集上的梯度。这种方法有效利用保留的历史信息，从而在联邦遗忘任务中实现更高的存储效率。

基于日志的事务回滚机制在数据库管理系统中是确保数据一致性和完整性的关键手段。其记录的所有操作日志在回滚时可逆向执行已完成的操作。受此机制启发，Yuan 等人[29]提出了一种有效的联邦遗忘学习方法 FRU，如图 10 所示，FRU 法通过回滚和校准历史参数更新消除参与者的贡献，并利用这些更新加速联邦推荐的重建。具体而言，FRU 使用以下公式回滚全局模型：

$$M(V)_k^{t+1}=\text{select}(\|M(V)_k^t\|) \tag{37}$$

其中，$M(V)_k^t$ 表示每个项嵌入更新的显著性，选择重要更新以减少存储空间。

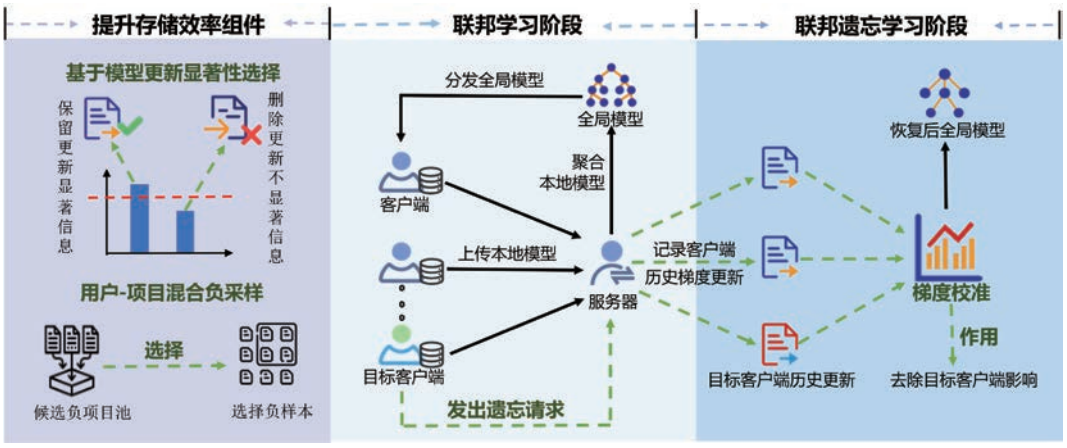

图 10　FRU 算法提升存储效率框架

为应对资源受限设备无法存储所有历史参数更新的挑战，该方法提出了用户-项目混合负采样和基于重要性的更新选择两个创新组件，用户-项目混合的负采样通过以下公式减少每个客户端的项目嵌入更新大小：

$$V_{\text{neg}}=\text{Random}(V_{\text{neg}})=N\times\beta(V_u\cup V_v) \tag{38}$$

其中，V_u 和 V_v 分别代表用户和项目的负样本集合，β 是一个缩放因子，用来减少每个用户需要的负样本数量。此外，基于重要性的更新选择组件在每个训练周期内动态选择重要更新进行存储，而非保存所有参数更新，存储空间成本可以通过以下公式减少：

$$C_{\text{storage}}=\beta\times B\times\alpha(|V_{\text{pos}}|+\beta|V_{\text{neg}}|)\times C \tag{39}$$

其中，α 和 β 分别是用于选择重要更新和调整负样本比例的因子，$|V_{\text{pos}}|$ 和 $|V_{\text{neg}}|$ 分别是正样本和负样本的数量。这两个组件显著提升了联邦遗忘学习过程中的存储效率，并解决了资源受限客户端无法参与共同训练的难题。

6.2　不依赖历史信息提升存储效率

不依赖历史信息的方案则通过压缩、分片或者选择性存储来提升存储效率。这类方法通常侧重于

在每轮训练中动态优化参数更新，避免长期存储大量历史数据。例如，Li 等人[43]提出的 FedAF 框架通过模仿神经科学中的主动遗忘过程来实现联邦遗忘学习，即使用新记忆（新的训练数据）覆盖旧记忆（需要遗忘的训练数据）来实现对目标数据样本的遗忘，从而无需存储大量历史更新，显著提升了算法的存储效率。

基于知识蒸馏方法的联邦遗忘学习方法 FUKD[26]要求服务器保留客户端更新的历史记录。此外，该方法还需服务器拥有额外的未标记数据，因此在存储效率方面尚需进一步优化。为了解决这一问题，Li 等人[58]提出的基于子空间的 SFU 方法引入了一种新的基于编码计算的分片机制。该方法允许全局模型在由其他客户端形成的输入梯度空间中存储未编码的中间模型参数，同时收集编码后的中间模型参数。每个客户端将编码后的参数分发给其他客户端，服务器通过共享的密钥访问这些切片，并通过解码过程重构原始模型参数。由于编码后的模型参数大小远小于未编码的参数，该方法显著减少了服务器的存储开销。

在联邦遗忘学习中，提升存储效率是应对资源受限设备存储能力有限的重要策略。为了有效地提升存储效率，如表 8 所示，研究者们探索了两种主要的策略：一方面，依赖历史信息的方法通过引入创新的存储和校准技术、知识蒸馏以及基于日志的回滚机制，能够在一定程度上提高存储效率。然而，这种方法也增加了隐私泄露的风险，尤其是在处理敏感数据时，可能导致用户信息的泄露；另一方面，不依赖历史信息的方案则通过压缩、覆盖或分片策略，减少长期存储需求。这种方法在一定程度上避免了隐私泄露风险，但提升存储效率相对较慢。因此，如何在提高存储效率和保护隐私之间找到平衡，或设计出新方案，以同时兼顾这两个目标，成为未来亟需解决的重要问题。

表 8 联邦遗忘学习算法的存储效率对比

提升存储效率类别	算法/框架名称	空间复杂度	提升存储效率的关键技术	缺点
依赖历史信息提升	FedEraser[24]	$O(K\times I\times P)$ I 表示客户端的更新信息	校准技术解决客户端模型参数更新中信息的前向耦合	依赖客户端参与、需要存储检查点
	FUKD[26]	$O(K\times T\times P)$ T 表示为遍历删除特定客户端所需轮数	知识蒸馏技术	依赖原始全局模型的预测结果、存储历史信息可能造成隐私泄露
	FRU[29]	$O(K\times(M+P))$ M 表示客户端嵌入大小	用户-项目混合的负采样组件、基于重要性的更新选择组件	由于数据的稀疏性和偏差性，负样本采样方法会带来额外的计算开销
	FedRecovery[34]	$O(K\times P)$	梯度残差量化、早期停止策略、高斯机制	早期停止策略可能会影响模型的准确性
	QUICKDROP[37]	$O(K\times(N+S+P))$ N 表示客户端本地的数据量，S 代表蒸馏数据集大小	数据集蒸馏技术、梯度匹配技术	蒸馏数据的质量略低于原始数据，导致模型性能略低
不依赖历史信息提升	FedAF[43]	$O(K\times(Q+R+P))$ Q 代表教师模型的数量，R 代表教师模型生成新记忆的大小	模仿神经科学中的主动遗忘过程	教师模型数量过多会引起计算成本的增加
	SFU[58]	$O(K\times N\times P)$ N 表示客户端本地的数据	基于编码计算的分片机制：编码计算与编码重构	编码计算技术会导致计算与通信开销的增加

注：K 表示客户端数量，P 表示模型参数。

7 联邦遗忘学习的相关应用

近年来，随着科研工作者对联邦遗忘学习的深入研究，该领域已经在隐私安全和算法效率等方面取得了显著突破。在实际应用方面，联邦遗忘学习在个性化推荐系统、数字孪生系统、在线排名学习系统和医疗保健系统等领域取得了一些成果。

7.1 个性推荐系统

随着数据隐私在推荐系统中的重要性日益凸显，联邦推荐系统成为了众多学者研究的焦点。尽管有些研究[79-80]尝试将机器遗忘学习应用于推荐系统，但主要集中于传统的集中式推荐，这类方法在学习过程中需访问所有训练数据，而这在联邦推荐系统中是不可行的。因此，为了满足数据隐私保护的需求，联邦推荐系统需要具备有效的遗忘机制，以确保参与者能够撤回自己的数据贡献。赋予用户“被遗忘权”不仅是法律合规的要求，如 GDPR 规定的数据可撤销性，同时也有助于提高系统的安全性和鲁棒性，减少因恶意客户端攻击或数据污染导致

的性能下降。

针对联邦推荐系统中的遗忘问题,Yuan 等人[29]受数据库管理系统中基于日志的事务回滚机制的启发,提出了一种创新性的遗忘学习方法—FRU。该方法通过回滚和校准历史参数更新,有效消除了参与者在联邦推荐系统中的贡献,并利用这些更新加速了联邦推荐功能的重建。为了验证 FRU 的有效性,研究团队选择了两种广泛使用的推荐器—NCF[81] 和 LightGCN[82],并在 MovieLens-100k 和 Steam-200k 数据集上进行了实验。实验结果显示,FRU 不仅能够消除被删除的恶意参与者的影响,而且在重建联邦推荐功能时,与从头开始训练的方法相比,速度提高了至少 7 倍,这一成果为联邦推荐系统的稳健性和效率提供了有力支持。

尽管 FRU 在联邦推荐系统中引入了高效的遗忘机制,但仍然面临一些挑战。首先,在非独立同分布(Non-IID)的场景下,如何在遗忘过程中最小化模型性能损失仍然是一个关键问题。其次,虽然 FRU 加速了模型重建,但其依赖历史梯度信息进行回滚,这可能会增加存储和计算开销,尤其是在大规模联邦训练任务中。最后,当前的遗忘方法主要关注用户级别的数据移除,而在实际应用中,部分用户可能仅希望删除特定时间段的交互数据或某些敏感行为数据,如何支持更细粒度的遗忘需求仍需进一步研究。

7.2　医疗保健系统

在医疗保健系统中,隐私保护对于处理敏感数据(如电子健康记录、患者病历)至关重要。然而,简单地从数据集中移除参与者数据,并不足以确保真正的隐私保护,因为已训练模型可能仍然保留部分被删除数据的信息,导致潜在的数据重建或推断风险。因此,联邦遗忘学习成为医疗领域数据隐私保护的核心技术之一。一个理想的联邦遗忘学习框架应当能够灵活适应不同的应用环境,如在跨单机环境中频繁删除客户数据子集,或在跨设备环境中彻底删除客户的本地数据集[83]。然而,目前关于联邦遗忘学习在医疗场景的研究仍然有限,使得医疗系统在对抗性攻击下更易受到威胁。

在此背景下,Pan 等人[84]提出了安全压缩多集聚合(Secure Compressed Multi-Aggregation,SCMA)框架,旨在解决聚类过程中遇到的稀疏安全联邦遗忘学习问题。该方法采用了 Reed-Solomon 编码[85],结合特殊评估点,在降低通信复杂度的同时增强了秘密共享协议的安全性,并成功证明了客户端通信成本仅为向量维度的对数级别。为了进一步验证 SCMA 在解除学习过程中的有效性,研究团队对其相较于完全重新训练的性能进行了理论分析,并在多个医疗数据集上进行了实验。实验结果表明,在聚类大小高度不均衡的情况下,SCMA 仍能保持良好的聚类性能。具体而言,与每次移除请求都需要执行局部与全局完全重新训练的 K-means++框架[86]相比,SCMA 在七个数据集上的平均速度提升约 84 倍。该方法在提升计算效率的同时,也能够保护患者隐私,符合医疗行业的合规要求,并适用于多种医疗保健场景,从而满足不同机构和用户的需求。

尽管 SCMA 框架在联邦遗忘学习中具有良好应用前景,但仍面临多项技术挑战。首先,医疗数据通常是非独立同分布的,不同医院或设备的数据差异可能影响遗忘过程的效果,特别是在数据不均衡的情况下,可能降低模型的泛化能力。其次,尽管 SCMA 优化了通信成本,但秘密共享和编码操作仍带来额外的计算与存储开销,如何在隐私保护、计算复杂度和通信开销之间取得平衡仍需探索。最后,联邦学习可能受到数据投毒、模型窃取等对抗性攻击的威胁,如何确保遗忘机制的安全性仍是一个关键挑战。

7.3　数字孪生系统

5G 和 6G 移动网络凭借其高速率、低延迟和大容量特性,极大推动了智能服务的普及。然而,这些网络的大规模、复杂性和易出错性使得在移动网络中训练、测试及更新全局模型变得异常昂贵。为应对这些挑战,学术界和工业界正在积极探索数字孪生解决方案,以减轻这些压力,为联邦学习模型的应用提供更优的环境。

数字孪生技术是一种高级仿真方法,通过集成物理模型、实时传感器数据和运行历史纪录等多源信息,构建与真实实体全生命周期过程相匹配的精确数字模型。在实际网络条件不可靠或操作环境受限的情况下,数字孪生移动网络能够为机器学习模型提供持续的训练和优化环境[87]。此外,在大规模部署之前,机器学习模型可以在数字孪生网络中进行全面测试,以确保其在真实环境中的性能和稳定性。

然而,在数字孪生系统中,数据来自多个分布式设备,这些设备的传感器持续采集并上传大量动态变化的实时数据。传统的联邦学习尽管通过将数据保留在本地设备进行训练,减少了数据泄露的风险,

但它无法主动管理或删除过时、不相关的数据，导致模型可能会在无效数据上进行学习，影响其性能和决策精度。联邦遗忘学习在此背景下显得尤为重要。它通过在联邦学习框架中引入数据遗忘机制，使系统能够动态剔除无效或过时数据，确保模型始终基于最新、最相关的信息进行优化，从而提升系统的整体智能化水平。首先，联邦遗忘学习能够满足数据隐私与合规性需求，允许用户主动请求删除其贡献的数据。其次，该机制优化了模型训练效率，减少存储和计算资源的浪费，加速模型收敛，同时降低无效数据对系统计算负担的影响。更重要的是，联邦遗忘学习能够提升模型的泛化能力和自适应性，确保模型不会受到历史数据的干扰，从而更精准地反映当前数字孪生系统所处的状态，提高预测与决策的准确性。因此，Xia 等人[88]针对隐私保护领域中的数据遗忘问题，提出了一种名为 FedME2 的遗忘框架。该框架由 MEval 和 MErase 两个核心模块构成。FedME2 利用 MEval 模块提供的记忆评价信息作为指导，结合 MErase 模块的多损失训练方法，在数字孪生网络环境中实现高效且精确的数据遗忘管理。研究团队在四个具有代表性的数字孪生网络虚拟环境中测试了 FedME2 的性能，结果显示，FedME2 框架在全局模型上的平均数据遗忘率达约 75%，且对全局模型精度的影响严格控制在 4%以下。这表明，FedME2 具备更好的数据遗忘功能，能够在保证模型准确性的同时，保护数字孪生网络的数据隐私。

尽管 FedME2 在数字孪生环境中表现出良好应用前景，但仍面临多重挑战。首先，数字孪生网络中的高频动态数据要求遗忘机制能够快速响应，以确保全局模型稳定性。其次，边缘设备计算能力有限，如何在资源受限条件下优化计算与通信成本仍待研究。此外，该方法需兼容不同类型的模型，以适应智能制造、自动驾驶等多场景需求。同时，联邦遗忘学习的计算开销较高，如何降低能耗以适应能量受限设备是未来优化方向。最后，不同地区的数据法规对隐私保护要求不同，实现灵活合规的数据遗忘机制仍是关键挑战。因此，未来需在数据动态适应、计算优化、模型兼容性、绿色计算及隐私合规性等方面进一步研究，以提升联邦遗忘学习的实际应用价值。

7.4 排名学习系统

在线排名学习（Online Learning to Rank，OLTR）框架内，排序模型通过实时接收参与者对查询与文档之间相关性的隐式反馈（如点击数据）不断更新其参数。然而，这一机制要求将参与者的交互数据上传至服务器以进行集中训练，从而引发参与者数据隐私问题。因此，联邦遗忘学习在在线排名学习领域展现出巨大的应用潜力。通过采用联邦遗忘学习策略，可以在保护参与者数据隐私的同时，实现模型的持续更新和优化。

将联邦遗忘学习应用于在线排名学习领域面临两大挑战：一是如何在不增加计算负担的前提下实现数据的高效遗忘；二是如何准确评估遗忘方法的有效性。为此，Wang 等人[89]将 FedEraser 方法引入在线排名学习的环境，并针对性地调整为在线训练模式。在 Wang 等人的方法中，历史更新信息被妥善保存在本地设备中，并在重新训练新的排序器时发挥关键作用。这种设计旨在降低额外计算成本，确保在有限资源下也能实现高效的遗忘。

为进一步评估模型性能，研究团队引入了投毒攻击方法。通过放大和控制客户端离开模型的影响，可以更准确地了解联邦遗忘学习在在线排名学习领域的实际应用效果，并为未来的优化提供有力数据支持。在线排名学习与传统分类任务在处理方式、在线学习与离线学习的实践，以及隐式参与者反馈与真实值标签的应用等方面存在明显区别。

尽管该方法在在线排名学习中展现了良好的应用价值，但仍然存在一定挑战。首先，在线排名学习依赖于实时的用户反馈，而联邦遗忘学习可能会影响数据的时效性和连续性，导致模型在更新过程中面临信息丢失或延迟的问题。其次，排序任务通常涉及复杂的交互模式，不同用户的点击行为可能受到个性化推荐、界面设计等因素的影响，这使得精准识别和遗忘特定用户贡献变得更加困难。此外，在资源受限的边缘设备上执行排序模型的遗忘学习，可能会带来额外的计算和存储开销，如何在保证高效性的同时减少资源占用仍需进一步优化。

7.5 投毒防御应用

联邦学习作为一种去中心化的机器学习方法，允许多个客户端在不共享原始数据的前提下，共同训练全局模型。然而，由于其去中心化特性，服务器难以直接控制和验证客户端上传的本地更新，这使得恶意客户端可以利用这一机制实施投毒攻击。投毒攻击主要分为数据投毒和模型投毒两种方式，前者通过篡改数据样本或标签影响模型学习，后者则直接修改本地训练模型的参数，使得聚合后的全局模型受到污染。

联邦遗忘学习可以在不重新训练整个模型的情况下,有效去除恶意客户端的影响,使全局模型恢复至干净状态。相比于完全从头训练,联邦遗忘学习能够以更低的计算成本实现模型修正,并减少不必要的数据处理。然而,当前联邦遗忘学习研究主要关注其效率和准确性,而在安全性方面仍然存在严重的隐患。研究发现,投毒攻击可以扩展至联邦遗忘学习阶段,即使服务器在遗忘过程中识别并移除了恶意客户端,攻击者仍可通过特定策略干扰服务器的执行,使得最终遗忘后的模型仍然保持投毒状态。

针对联邦遗忘学习阶段的安全性问题,Wang等人[90]提出了 UnlearnGuard 作为防御机制,以增强联邦遗忘学习对投毒攻击的鲁棒性。UnlearnGuard 的核心思想是利用历史数据存储、模型更新预测和过滤策略相结合,以确保遗忘学习过程中不会受到恶意更新的干扰。首先,服务器在联邦学习阶段存储每个客户端的本地模型更新历史记录,这些数据可以用于在联邦遗忘学习阶段估计合理的模型更新。其次,服务器利用 L-BFGS 算法近似计算 Hessian 矩阵,从而预测遗忘学习过程中可能出现的模型更新,并进行合理性判断。最后,UnlearnGuard 通过两种不同的筛选策略进一步提高防御能力:基于距离筛选(UnlearnGuard-Dist)确保模型更新不会偏离历史模式,而基于方向筛选(UnlearnGuard-Dir)则进一步优化筛选机制,不仅关注更新的大小,还考虑其方向,以防止恶意小幅调整绕过筛选。

总之,随着人工智能的不断发展,数据的隐私保护也变得更为重要。研究人员可以致力于设计更有效的联邦遗忘学习方法,提高模型的遗忘能力和稳定性。未来的研究和发展将进一步推动联邦遗忘学习的应用探索,联邦遗忘学习也会在更多领域发挥其重要作用,助力实现安全可信的人工智能。

8　总结与展望

本文聚焦于联邦遗忘学习中的隐私保护、模型恢复、计算效率以及存储效率四个方面,详细梳理了联邦遗忘学习现存问题以及解决方案。在隐私保护层面,本文从服务器和客户端两个角度详细分析了联邦遗忘学习面临的隐私风险及相应的解决方案;在模型性能恢复层面,本文深入剖析了残留遗忘数据的影响以及灾难性遗忘对联邦遗忘学习模型性能的影响,并总结了相应的解决方法;在计算效率层面,本文将优化算法分为基于历史信息的遗忘和基于快速重新训练的两类,并对相关算法与经典算法进行了比较和总结;同时,根据是否依赖历史信息对提升存储效率的方案进行了分类和深入地对比分析。此外,本文还总结了关于联邦遗忘学习的实际应用,以促进这一领域的发展与创新。

在未来的研究工作中,联邦遗忘学习仍存在一些亟需解决的挑战:

(1)计算效率与隐私保护难权衡。在联邦遗忘学习中,基于历史信息恢复全局模型通常比基于快速重新训练的方法更高效,但由于需要存储参与者的历史信息,这种方法可能面临隐私泄露的风险。在考虑实际的遗忘优化算法时,应当权衡计算效率与隐私保护两方面关键因素寻找其平衡点。此外,应当根据具体应用场景和需求,调整计算效率和隐私保护的权重,例如在隐私敏感场景中提高隐私保护的权重,而在计算效率要求高的场景中提高计算效率的权重。

(2)防御机制与现有方案不兼容。现有的防御机制,如安全多方计算、秘密共享等,在联邦学习的隐私保护中取得了一定成效,但是这些方法通常与联邦遗忘学习框架和优化算法难以兼容,无法直接集成来达到抵御攻击的效果。要解决这一问题,需要在算法设计和系统框架层面进行协调,也可以重新设计联邦遗忘学习的系统架构,以更好地支持各种防御机制的集成。同时,可以开发更加轻量级和高效的隐私保护机制,以降低与现有方案的兼容成本。如利用联邦生成对抗网络的思想,设计出更加高效的联邦差分隐私方法。

(3)模型恢复速度与模型恢复精度难兼顾。在联邦遗忘学习的计算效率方面,恢复全局模型的方式可以分为基于历史信息和基于快速重新训练两类。基于历史信息的遗忘方法通常比基于快速重新训练的遗忘方法更为高效,因为它舍弃了一定的模型精度,以近似恢复全局模型;而快速重新训练则可以达到与原模型高度相近的模型精度。在未来的研究中,应根据实际需求动态调整恢复速度与精度之间的权重,或采用多目标优化策略,通过权衡和优化寻找速度和精度的最佳平衡点。

(4)缺乏激励机制导致参与者易退出。在联邦遗忘学习中,参与者可能因频繁的遗忘和恢复操作选择退出,从而影响系统的稳定性。例如,当系统中频繁进行遗忘和恢复操作时,其他参与者可能会耗

费额外计算与存储成本,从而导致其他参与者退出。通过设计合适的激励机制,可以鼓励有价值的参与者留在系统中,从而维持平台的稳定性和可持续性。除此之外,激励机制可以激励参与者积极参与联邦遗忘学习过程,以提高恢复模型准确性和效用。然而,目前关于设计联邦遗忘学习激励机制的研究仍然不够充分。在未来的研究中,可以为参与者提供明确的奖励机制或者根据参与者的参与历史和贡献情况引入声誉系统给予声誉评分,以激励参与者长期参与。

(5)频繁的遗忘请求难以快速响应。在实际的联邦遗忘学习中可能会发生频繁的遗忘请求,但是服务器可能会因繁杂的处理请求从而导致其发生处理紊乱的情况。现有的遗忘方案大多集中于处理单个遗忘请求,如何高效地处理多个遗忘请求并快速地进行模型的恢复则是接下来应当研究的方向。可以考虑将多个遗忘请求批量处理,减少每次处理的开销,提高系统的响应速度。通过实现负载均衡机制,将遗忘请求均匀分配到不同的计算节点,避免单个节点过载,提高整体系统的处理效率。

在未来的研究工作中,应更好地完善联邦遗忘学习隐私安全和算法效率,以促进联邦学习的应用安全,加快分布式机器学习的发展。

致 谢 感谢《计算机学报》编辑和审稿专家,他们付出了辛勤工作。

参 考 文 献

[1] Li H, Yu L, He W. The impact of GDPR on global technology development. Journal of Global Information Technology Management, 2019, 22(1): 1-6

[2] Harding E L, Vanto J J, Clark R, et al. Understanding the scope and impact of the California consumer privacy act of 2018. Journal of Data Protection & Privacy, 2019, 2(3): 234-253

[3] McMahan B, Moore E, Ramage D, et al. Communication-efficient learning of deep networks from decentralized data//Proceedings of the 20th International Conference on Artificial Intelligence and Statistics. Fort Lauderdale, USA, 2017: 1273-1282

[4] Kairouz P, McMahan H B, Avent B, et al. Advances and open problems in federated learning. Foundations and Trends in Machine Learning, 2021, 14(1-2): 1-210

[5] Lo S K, Lu Q, Wang C, et al. A systematic literature review on federated machine learning: From a software engineering perspective. ACM Computing Surveys (CSUR), 2021, 54(5): 1-39

[6] Bhagoji A N, Chakraborty S, Mittal P, et al. Analyzing federated learning through an adversarial lens//Proceedings of the 36th International Conference on Machine Learning. Long Beach, USA, 2019: 634-643

[7] Liu J, Xue M, Lou J, et al. Muter: Machine unlearning on adversarially trained models//Proceedings of the IEEE/CVF International Conference on Computer Vision. Paris, France, 2023: 4892-4902

[8] Chourasia R, Shah N. Forget unlearning: Towards true data-deletion in machine learning//Proceedings of the 40th International Conference on Machine Learning. Honolulu, USA, 2023: 6028-6073

[9] Cao Y, Yang J. Towards making systems forget with machine unlearning//Proceedings of the 2015 IEEE Symposium on Security and Privacy. San Jose, USA, 2015: 463-480

[10] Varoquaux G, Colliot O. Evaluating machine learning models and their diagnostic value. Machine Learning for Brain Disorders, 2023: 601-630

[11] GRAVES L, NAGISETTY V, GANESH V. Amnesiac machine learning//Proceedings of the AAAI Conference on Artificial Intelligence. Vancouver, Canada, 2022: 11516-11524

[12] Bourtoule L, Chandrasekaran V, Choquette-Choo C A, et al. Machine unlearning//Proceedings of the 2021 IEEE Symposium on Security and Privacy (SP). San Francisco, USA, 2021: 141-159

[13] Liu G, Ma X, Yang Y, et al. Federated unlearning. arXiv preprint arXiv:2012.13891, 2020

[14] Yang J, Zhao Y. A Survey of Federated Unlearning: A Taxonomy, Challenges and Future Directions. arXiv preprint arXiv:2310.19218, 2023

[15] Wang F, Li B, Li B. Federated unlearning and its privacy threats. IEEE Network, 2023,38(2): 294-300

[16] Liu Z, Jiang Y, Shen J, et al. A survey on federated unlearning: Challenges, methods, and future directions. arXiv preprint arXiv:2310.20448, 2023

[17] Romandini N, Mora A, Mazzocca C, et al. Federated unlearning: A survey on methods, design guidelines, and evaluation metrics. arXiv preprint arXiv:2401.05146, 2024

[18] Wang Peng-Fei, Wei Zong-Zheng, Zhou Dong-Sheng, et al. A survey on federated unlearning. Chinese Journal of Computers, 2024,47(02):396-422 (in Chinese)
(王鹏飞,魏宗正,周东生等.联邦忘却学习研究综述.计算机学报,2024,47(02):396-422)

[19] Kurmanji M, Triantafillou P, Hayes J, et al. Towards unbounded machine unlearning. Advances in Neural Information Processing Systems, 2024: 1957-1987

[20] Nguyen T H, Vu H P, Nguyen D T, et al. Empirical study of federated unlearning: Efficiency and effectiveness//Proceedings of the 15th Asian Conference on Machine Learning. Istanbul, Türkiye, 2024: 959-974

[21] Chundawat V S, Tarun A K, Mandal M, et al. Zero-shot

machine unlearning. IEEE Transactions on Information Forensics and Security, 2023: 2345-2354

[22] Baumhauer T, Schöttle P, Zeppelzauer M. Machine unlearning: Linear filtration for logit-based classifiers. Machine Learning, 2022, 111(9): 3203-3226

[23] Foster J, Schoepf S, Brintrup A. Fast machine unlearning without retraining through selective synaptic dampening//Proceedings of the AAAI Conference on Artificial Intelligence. Vancouver, Canada, 2024: 12043-12051

[24] Liu G, Ma X, Yang Y, et al. Federaser: Enabling efficient client-level data removal from federated learning models//Proceedings of the 29th International Symposium on Quality of Service (IWQOS). Tokyo, Japan, 2021: 1-10

[25] Wang J, Guo S, Xie X, et al. Federated unlearning via class-discriminative pruning//Proceedings of the ACM Web Conference 2022. Lyon, France, 2022: 622-632

[26] Wu C, Zhu S, Mitra P. Federated unlearning with knowledge distillation. arXiv preprint arXiv:2201.09441, 2022

[27] Gao X, Ma X, Wang J, et al. Verifi: Towards verifiable federated unlearning. arXiv preprint arXiv:2205.12709, 2022

[28] Wu L, Guo S, Wang J, et al. Federated unlearning: Guarantee the right of clients to forget. IEEE Network, 2022, 36(5): 129-135

[29] Yuan W, Yin H, Wu F, et al. Federated unlearning for on-device recommendation//Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining. Singapore, 2023: 393-401

[30] Zhu X, Li G, Hu W. Heterogeneous federated knowledge graph embedding learning and unlearning//Proceedings of the ACM Web Conference 2023. Austin, USA, 2023: 2444-2454

[31] Wu C, Zhu S, Mitra P, et al. Unlearning backdoor attacks in federated learning//Proceedings of the 2024 IEEE Conference on Communications and Network Security (CNS). Taipei, China, 2024: 1-9

[32] Jin R, Chen M, Zhang Q, et al. Forgettable federated linear learning with certified data removal. arXiv preprint arXiv: 2306.02216, 2023

[33] Che T, Zhou Y, Zhang Z, et al. Fast federated machine unlearning with nonlinear functional theory//Proceedings of the 40th International Conference on Machine Learning. Honolulu, USA, 2023: 4241-4268

[34] Zhang L, Zhu T, Zhang H, et al. Fedrecovery: Differentially private machine unlearning for federated learning frameworks. IEEE Transactions on Information Forensics and Security, 2023: 4732-4746

[35] Fraboni Y, Van Waerebeke M, Scaman K, et al. Sequential informed federated unlearning: Efficient and provable client unlearning in federated optimization. arXiv preprint arXiv: 2211.11656, 2022

[36] Zhao Y, Wang P, Qi H, et al. Federated unlearning with momentum degradation. IEEE Internet of Things Journal, 2023,11(5): 8860-8870

[37] Dhasade A, Ding Y, Guo S, et al. QuickDrop: Efficient federated unlearning by integrated dataset distillation. arXiv preprint arXiv:2311.15603, 2023

[38] Xiong Z, Li W, Li Y, et al. Exact-fun: An exact and efficient federated unlearning approach//Proceedings of the 2023 IEEE International Conference on Data Mining (ICDM). Shanghai, China, 2023: 1439-1444

[39] Lin Y, Gao Z, Du H, et al. Scalable federated unlearning via isolated and coded sharding. arXiv preprint arXiv: 2401.15957, 2024

[40] Lin Y, Gao Z, Du H, et al. Blockchain-enabled trustworthy federated unlearning. arXiv preprint arXiv:2401.15917, 2024

[41] Liu Y, Xu L, Yuan X, et al. The right to be forgotten in federated learning: An efficient realization with rapid retraining//Proceedings of the IEEE INFOCOM 2022-IEEE Conference on Computer Communications. London, UK, 2022: 1749-1758

[42] Su N, Li B. Asynchronous federated unlearning//Proceedings of the IEEE INFOCOM 2023-IEEE Conference on Computer Communications. Hoboken, USA, 2023: 1-10

[43] Li Y, Chen C, Zheng X, et al. Federated unlearning via active forgetting. arXiv preprint arXiv:2307.03363, 2023

[44] Halimi A, Kadhe S, Rawat A, et al. Federated unlearning: How to efficiently erase a client in fl?. arXiv preprint arXiv: 2207.05521, 2022

[45] Li X, Huang K, Yang W, et al. On the convergence of fedavg on non-iid data. arXiv preprint arXiv:1907.02189, 2019

[46] Chen Y, Ning Y, Slawski M, et al. Asynchronous online federated learning for edge devices with non-iid data//Proceedings of the 2020 IEEE International Conference on Big Data (Big Data). Atlanta, USA, 2020: 15-24

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

[48] Fang M, Cao X, Jia J, et al. Local model poisoning attacks to Byzantine-Robust federated learning//Proceedings of the 29th USENIX security symposium (USENIX Security 20). Boston, USA, 2020: 1605-1622

[49] Golatkar A, Achille A, Ravichandran A, et al. Mixed-privacy forgetting in deep networks//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. Nashville, USA, 2021: 792-801

[50] Yang W, Wang N, Guan Z, et al. A practical cross-device federated learning framework over 5g networks. IEEE Wireless Communications, 2022, 29(6): 128-134

[51] Liu Y, Ma Z, Liu X, et al. Learn to forget: User-level memorization elimination in federated learning. arXiv preprint arXiv:2003.10933, 2020

[52] Tan K Y, Lyu Y, Ong Y S, et al. Unfolded self-reconstruction LSH: Towards machine unlearning in approximate nearest neighbour search. arXiv preprint arXiv:2304.02350, 2023

[53] Shaik T, Tao X, Li L, et al. Framu: Attention-based machine unlearning using federated reinforcement learning. IEEE Transactions on Knowledge and Data Engineering, 2024,36(10): 1-14

[54] Golatkar A, Achille A, Soatto S. Eternal sunshine of the spotless net: Selective forgetting in deep networks//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, USA, 2020: 9304-9312

[55] Posocco N, Bonnefoy A. Estimating expected calibration errors//Proceedings of the 30th International Conference on Artificial Neural Networks. Bratislava, Slovakia, 2021: 139-150

[56] Fredrikson M, Jha S, Ristenpart T. Model inversion attacks that exploit confidence information and basic countermeasures//Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. Denver, USA, 2015: 1322-1333

[57] Gao W, Zhang X, Guo S, et al. Automatic transformation search against deep leakage from gradients. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023,45(9): 10650-10668

[58] Li G, Shen L, Sun Y, et al. Subspace based Federated Unlearning. arXiv preprint arXiv:2302.12448, 2023

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

[60] Nasr M, Shokri R, Houmansadr A. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning//Proceedings of the 2019 IEEE symposium on security and privacy (SP). San Francisco, USA, 2019: 739-753

[61] Yang H, Ge M, Xiang K, et al. Using highly compressed gradients in federated learning for data reconstruction attacks. IEEE Transactions on Information Forensics and Security, 2022, 18: 818-830

[62] Ma J, Naas S A, Sigg S, et al. Privacy-preserving federated learning based on multi-key homomorphic encryption. International Journal of Intelligent Systems, 2022, 37(9): 5880-5901

[63] Zhang C, Li S, Xia J, et al. BatchCrypt: Efficient homomorphic encryption for Cross-Silo federated learning//Proceedings of the 2020 USENIX annual technical conference (USENIX ATC 20). Boston, USA, 2020: 493-506

[64] Mazzone F, van den Heuvel L, Huber M, et al. Repeated knowledge distillation with confidence masking to mitigate membership inference attacks//Proceedings of the 15th ACM Workshop on Artificial Intelligence and Security. Los Angeles, USA, 2022: 13-24

[65] Cheng A, Wang P, Zhang X S, et al. Differentially private federated learning with local regularization and sparsification//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. New Orleans, USA, 2022: 10122-10131

[66] Shokri R, Stronati M, Song C, et al. Membership inference attacks against machine learning models//Proceedings of the 2017 IEEE symposium on Security and Privacy (SP). San Jose, USA, 2017: 3-18

[67] Liu Y, Fan M, Chen C, et al. Backdoor defense with machine unlearning//Proceedings of the IEEE INFOCOM 2022-IEEE Conference on Computer Communications. London, UK, 2022: 280-289

[68] Van de Ven G M, Tuytelaars T, Tolias A S. Three types of incremental learning. Nature Machine Intelligence, 2022, 4(12): 1185-1197

[69] Liu Y, Ma Z, Yang Y, et al. Revfrf: Enabling cross-domain random forest training with revocable federated learning. IEEE Transactions on Dependable and Secure Computing, 2021, 19(6): 3671-3685

[70] Alam M, Lamri H, Maniatakos M. Get rid of your trail: Remotely erasing backdoors in federated learning. arXiv preprint arXiv:2304.10638, 2023

[71] Wang H, Zhu X, Chen C, et al. Goldfish: An efficient federated unlearning framework. arXiv preprint arXiv:2404.03180, 2024

[72] Mohamadi M A, Bae W, Sutherland D J. A fast, well-founded approximation to the empirical neural tangent kernel//Proceedings of the 40th International Conference on Machine Learning. Honolulu, USA, 2023: 25061-25081

[73] Zhang Z, Zhou Y, Zhao X, et al. Prompt certified machine unlearning with randomized gradient smoothing and quantization. Advances in Neural Information Processing Systems, 2022, 35: 13433-13455

[74] Grube S. Strong solutions to McKean-Vlasov SDEs with coefficients of Nemytskii-type. Electronic Communications in Probability, 2023, 28: 1-13

[75] Yue Y, Jiang J, Ye Z, et al. Sharpness-aware minimization revisited: Weighted sharpness as a regularization term//Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. Long Beach, USA, 2023: 3185-3194

[76] Verma A, Bhattacharya P, Bodkhe U, et al. FedRec: Trusted rank-based recommender scheme for service provisioning in federated cloud environment. Digital Communications and Networks, 2023, 9(1): 33-46

[77] Cui L, Su X, Zhou Y. A fast blockchain-based federated learning framework with compressed communications. IEEE Journal on Selected Areas in Communications, 2022, 40(12): 3358-3372

[78] Jeong H, Ma S, Houmansadr A. SoK: Challenges and opportunities in federated unlearning. arXiv preprint arXiv: 2403.02437, 2024

[79] Chen C, Sun F, Zhang M, et al. Recommendation unlearning//Proceedings of the ACM Web Conference 2022. Lyon, France, 2022: 2768-2777

[80] Li Y, Chen C, Zheng X, et al. Making recommender sys-

tems forget: Learning and unlearning for erasable recommendation. Knowledge-Based Systems, 2024, 283: 111124

[81] He X, Liao L, Zhang H, et al. Neural collaborative filtering//Proceedings of the 26th international conference on world wide web. Perth, Australia, 2017: 173-182

[82] He X, Deng K, Wang X, et al. Lightgcn: Simplifying and powering graph convolution network for recommendation// Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval. Xi'an, China, 2020: 639-648

[83] Gu X, Sabrina F, Fan Z, et al. A review of privacy enhancement methods for federated learning in healthcare systems. International Journal of Environmental Research and Public Health, 2023, 20(15): 6539

[84] Pan C, Sima J, Prakash S, et al. Machine unlearning of federated clusters. arXiv preprint arXiv:2210.16424, 2022

[85] Yu L, Lin S J, Hou H, et al. Reed-solomon coding algorithms based on reed-muller transform for any number of parities. IEEE Transactions on Computers, 2023: 2677-2688

[86] Beretta L, Cohen-Addad V, Lattanzi S, et al. Multi-Swap k-Means++. Advances in Neural Information Processing Systems, 2024: 26069-26091

[87] Jafari M, Kavousi-Fard A, Chen T, et al. A review on digital twin technology in smart grid, transportation system and smart city: Challenges and future. IEEE Access, 2023, 11: 17471-17484

[88] Xia H, Xu S, Pei J, et al. Fedme 2: Memory evaluation & erase promoting federated unlearning in dtmn. IEEE Journal on Selected Areas in Communications, 2023: 3573-3588

[89] Wang S, Liu B, Zuccon G. How to Forget Clients in Federated Online Learning to Rank? //Proceedings of the European Conference on Information Retrieval. Glasgow, UK, 2024: 105-121

[90] Wang W, Ma Q, Zhang Z, et al. Poisoning Attacks and Defenses to Federated Unlearning. arXiv preprint arXiv:2501.17396, 2025

TANG Xiang-Yun, Ph. D., associate professor. Her main research interests include artificial intelligence security, federated learning, data security, and privacy protection.

WANG Wei, M. S. candidate. His main research interest is federated learning.

WENG Yu, Ph. D., professor. His main research interests include big data analysis, artificial intelligence, and deep learning.

SHEN Meng, Ph. D., professor. His main research interests include data security, artificial intelligence security, and blockchain security.

ZHANG Tao, Ph. D., associate professor. His main research interests include IoT security, and moving target defense.

WANG Wei, Ph. D., professor. His main research interests include network and system security, blockchain and privacy computing theory and technology.

ZHU Lie-Huang, Ph. D., professor. His main research interests include cryptozoologic algorithms and security protocols, blockchain technology, cloud computing security.

Background

This paper delves into Federated Unlearning (FU), focusing on the pressing need for "forgetting rights" in real-world applications. This concept involves removing contributions from clients or data samples in a trained global model, driven by privacy and security concerns. Existing data protection legislations like the General Data Protection Regulation (GDPR) and the California Consumer Privacy Act (CCPA) provide individuals with the right to request the deletion of their private data, emphasizing the importance of data privacy and user control. Currently, research efforts have explored Machine Unlearning techniques, enabling the removal of client or data sample contributions from global models. However, directly applying Machine Unlearning in Federated Learning scenarios faces challenges due to varying data distributions and features among participants, potentially compromising global model performance when attempting to forget specific clients' contributions. In response to these challenges, Federated Unlearning (FU) emerges as a novel approach to address forgetting requests in Federated Learning, aiming to erase target client or sample data contributions while maintaining model utility akin to starting training afresh. However, existing FU methodologies still face challenges in ensuring privacy protection and improving algorithm efficiency.

This survey reveals the core challenges faced by federated unlearning, systematically analyzes the implementation and challenges of privacy protection strategies, discusses in detail the solutions and effectiveness of secure forgetting, evaluates optimization strategies for computational and storage efficiency, and reviews the existing practical application scenarios of federated unlearning. The aim is to promote the development of this field through a comprehensive review and analytical discussion.

This project was supported by the National Natural Science Foundation of China Youth Fund (62302539, 62402029), the National Natural Science Foundation of China Joint Fund Key Project (U23A20304), and the China Postdoctoral Science Foundation (2024T170047, GZC20230223, 2024M750165).