电 子 科 技 大 学

UNIVERSITY OF ELECTRONIC SCIENCE AND TECHNOLOGY OF CHINA

硕士学位论文

MASTER THESIS

论文题目 **基于区块链的去中心化隐私增强联邦学习方案研究**

学科专业	软件工程
学　　号	**202221090117**
作者姓名	刘　悦
指导教师	赵洋　　副教授
学　　院	信息与软件工程学院

分类号 TP311.5 密级 公开

UDC 注1 004.41

学 位 论 文

基于区块链的去中心化隐私增强联邦学习方案研究

（题名和副题名）

刘 悦

（作者姓名）

指导教师 赵洋 副教授

电子科技大学 成 都

（姓名、职称、单位名称）

申请学位级别 硕士 学科专业 软件工程

提交论文日期 2025 年 4 月 8 日 论文答辩日期 2025 年 5 月 8 日

学位授予单位和日期 电子科技大学 2025 年 6 月

答辩委员会主席 张凤荔

评阅人 张凤荔、丁熠、陈波、于永斌、朱国斌

注 1：注明《国际十进分类法 UDC》的类号。

Research on Decentralized Privacy Enhanced Federated Learning Scheme Based on Blockchain

A Master Thesis Submitted to

University of Electronic Science and Technology of China

Discipline **Software Engineering**

Student ID **202221090117**

Author **Liu Yue**

Supervisor **Assoc. Prof. Zhao Yang**

School **School of Information and Software Engineering**

摘 要

为了解决传统联邦学习存在的单点信任风险、交互能力有限以及贡献评估不透明等问题，本文提出了一种基于联盟链的去中心化联邦学习框架。该框架通过多方协作机制与贡献量化方案，在确保过程可追溯性的同时实现激励分配的动态平衡。区块链的引入增强了系统的鲁棒性，但在梯度上传与链上存证环节增加了额外的通信开销。当面对设备算力约束和海量数据训练场景时，如何降低去中心化联邦学习的通信开销，成为亟待解决的问题。

在现有降低通信开销的解决方案中，主要依赖于梯度量化和稀疏传输。但这些方法通常基于静态压缩规则，并未充分考虑长期的传输均衡和信息的损失补偿。另一方面，现有的联邦学习贡献评估大多基于明文数据或训练过程日志，在数据加密条件下缺乏有效评估手段，因此无法满足评估过程中的隐私保护需求。此外，缺乏基于贡献度的正相关激励规则，易导造成高贡献度节点的边际收益递减，进而影响其参与积极性，最终制约模型的性能提升。为此，本文针对上述问题开展了研究，具体工作贡献如下：

1、针对传统联邦学习依赖中心化服务器带来的单点失效问题，设计了基于联盟链的去中心化联邦学习框架。该框架通过节点角色划分与动态轮选机制，构建了领导人节点、验证委员会节点与贡献评估委员会节点组成的多角色协作体系，并通过链上存证增强训练过程的透明性与可追溯性，为去中心化联邦学习提供了一种高鲁棒性的系统架构。

2、针对去中心化联邦学习的高通信开销问题，提出了一种基于动态误差感知的梯度压缩机制。该机制结合 Rand-k 稀疏采样与误差补偿技术，并采用 Paillier 加密保护梯度数据，兼顾了通信效率、模型收敛速度与数据隐私保护的需求，可显著降低梯度上传与链上存储的开销，为通信受限场景下的联邦学习提供了一种可行的梯度压缩方案。

3、针对贡献评估与激励机制缺失的问题，提出了一种基于长期公平性的贡献评估机制。该机制基于加密余弦相似度计算客户端贡献度，并结合历史贡献衰减与节点冷却轮换策略，动态评估并记录每个客户端的长期贡献表现，并将贡献评估结果上链，形成可追溯的贡献履历与透明激励规则，提升了系统整体的公平性与激励效果。

关键词：联邦学习，联盟链，梯度压缩，贡献评估，激励机制

ABSTRACT

In order to address the issues of single point of trust risk, limited interaction capability, and opaque contribution evaluation in traditional federated learning, this thesis proposes a decentralized federated learning framework based on consortium chains. This framework achieves dynamic balance in incentive allocation while ensuring process traceability through multi-party collaboration mechanisms and contribution quantification schemes. The introduction of blockchain enhances the robustness of the system, but adds additional communication overhead in the gradient upload and on chain certificate storage stages. When faced with device computing power constraints and massive data training scenarios, how to reduce the communication overhead of decentralized federated learning has become an urgent problem to be solved.

In existing solutions to reduce communication overhead, gradient quantization and sparse transmission are mainly relied upon. However, these methods are usually based on static compression rules and do not fully consider long-term transmission balance and information loss compensation. On the other hand, existing federated learning contribution evaluations are mostly based on plaintext data or training process logs, lacking effective evaluation methods under data encryption conditions, and therefore cannot meet the privacy protection needs during the evaluation process. In addition, the lack of positive correlation incentive rules based on contribution can lead to a decrease in marginal returns for high contribution nodes, which in turn affects their participation enthusiasm and ultimately constrains the performance improvement of the model. Therefore, this article has conducted research on the above-mentioned issues, and the specific contributions of the work are as follows:

1. A decentralized federated learning framework based on consortium chains has been designed to address the single point of failure problem caused by traditional federated learning relying on centralized servers. This framework constructs a multi role collaborative system consisting of leadership nodes, validation committee nodes, and contribution evaluation committee nodes through node role partitioning and dynamic rotation mechanism. It enhances the transparency and traceability of the training process

through on chain evidence, providing a highly robust system architecture for decentralized federated learning.

2. A gradient compression mechanism based on dynamic error perception is proposed to address the high communication overhead problem of decentralized federated learning. This mechanism combines Rand-k sparse sampling and error compensation techniques, and uses Paillier encryption to protect gradient data, balancing the requirements of communication efficiency, model convergence speed, and data privacy protection. It can significantly reduce the overhead of gradient upload and on chain storage, providing a feasible gradient compression scheme for federated learning in communication limited scenarios.

3. A contribution evaluation mechanism based on long-term fairness is proposed to address the issue of missing contribution evaluation and incentive mechanisms. This mechanism calculates client contribution based on encrypted cosine similarity, and combines historical contribution decay and node cooling rotation strategies to dynamically evaluate and record the long-term contribution performance of each client. The contribution evaluation results are then uploaded to the chain to form a traceable contribution history and transparent incentive rules, improving the overall fairness and incentive effect of the system.

Keywords: Federated Learning, Consortium Blockchain, Gradient Compression, Contribution Evaluation, Incentive Mechanism

目 录

图目录

表目录

第一章 绪论

1.1 研究背景及意义

近年来，深度学习技术的发展显著提升了人工智能的应用潜力，根据 ImageNet 竞赛的成果反映：模型的良好表现往往依赖于大量标注数据的训练基础[1]。但在现实情形中，数据生态正面临着艰巨的考验，比如在医疗诊断、金融风控、工业物联网等关键领域，各类机构出于对隐私合规要求以及商业利益考量，大多存在数据开放障碍，这种“数据孤岛”现象造成了单一主体获取充足训练样本时面临难题，显著制约了人工智能模型在实际业务时的泛化能力[2]。

更严峻的是，传统集中式机器学习要求将分散数据汇聚至中心服务器，这与《个人信息保护法》等法规产生根本性冲突。2023 年欧盟人工智能法案明确规定，未经明确授权的数据聚合行为将面临高额处罚[3]。这种数据可用性与隐私保护之间的矛盾，成为制约 AI 发展的“哥德巴赫猜想”。

为解决上述矛盾，联邦学习（Federated Learning, FL）开创了“数据不动模型动”的分布式学习范式[4]。其核心在于：各参与方在本地训练模型，仅通过加密方式交换模型参数而非原始数据。这种机制成功规避了数据跨境传输的法律风险，在金融、医疗与教育等领域展现出不可替代的价值。在金融领域，基于 FATE 框架实施纵向联邦学习的 FL-LRBC 模型通过整合商业银行与第三方支付机构的跨域特征，在 400 万样本规模下实现联邦信用评分建模。使信用评分模型测试 AUC 从单一机构基准的 0.69 提升至 0.76[5]；在医疗领域，来自宾夕法尼亚大学医学中心、宾夕法尼亚大学医学院和英特尔实验室的研究人员使用联邦学习技术开发了一种模型，该模型可以在磁共振成像图像中检测胶质母细胞瘤肿瘤，准确率较传统方法提升 33%[6]；教育场景中，联邦教育数据分析框架 FEEDAN 通过整合学生答题行为和辍学记录，在保护隐私的前提下，帮助教师识别学习困难学生，调整教学方法从而降低辍学率[7]。

联邦学习相较于传统集中式学习，开创了隐私安全与数据价值协同释放的先河，其“数据可用不可见”的核心范式成功化解了数据要素流通与个人权益保护间的根本矛盾[8]。然而，这一革新性框架在深入应用过程中显露出三重结构性困境：

（1）中心化聚合机制衍生的单点信任风险仍未根除[9]。传统联邦学习虽然避免了原始数据直接汇聚至中心服务器的传统模式，但在实际运行中，中心服务器仍承担着全局模型聚合与参数广播的关键角色。这种依赖中心节点的结构，使其天然成为系统中的“信任单点”，不仅聚焦了潜在攻击面，而且一旦该服务器遭遇

入侵、劫持或配置篡改，整个联邦网络的模型安全和性能都会受到系统性冲击。这种风险实质上是一种“结构性隐患”，难以通过单纯的加密或隐私增强技术彻底消解。

（2）审计凭证缺失导致多方责任界定困难[10]。传统联邦学习强调数据不出本地，但各参与方对模型训练的具体贡献、参数更新的质量、以及对全局性能的真实影响，往往缺乏透明可追溯的记录和标准化的凭证存档。尤其是在异构数据环境下，如果某一轮训练后的模型性能异常下滑，或者全局模型出现偏移甚至恶意后门嵌入，想要还原问题来源并界定责任方，往往面临“数据黑箱化”和“责任模糊化”的双重困境。缺乏过程级别的审计凭证，不仅降低了联邦参与方的信任感，也阻碍了事后责任认定与争议解决的效率。

（3）节点贡献失衡制约联邦协作生态活力[11]。传统联邦学习的目标是多方协同、共建共享，但现实中各参与节点的基础数据规模、数据质量、算法能力和算力条件存在巨大差异。高质量节点承担了主要贡献，而低质量节点的贡献极其有限，甚至部分“搭便车”节点纯粹享受联邦带来的好处而几乎无贡献。这种贡献-收益的严重失衡，直接影响了高价值节点的长期参与动力，甚至可能导致关键节点退出。缺乏动态评估贡献的机制，以及与贡献度强关联的差异化激励策略，进一步削弱了联邦学习网络的可持续性和协作活力。

上述困境的根源在于传统联邦架构存在信任锚点单一、审计不透明与贡献评估不准确三大系统性缺陷，而区块链联邦学习（Blockchain Federated Learning, BFL）通过架构层革新，系统性破解上述困境，推动联邦学习向可信化、市场化、合规化方向演进。它的架构更新如下：

（1）去中心化聚合重构信任体系[12]。BFL 在架构层面引入去中心化的账本网络，将原本由中心服务器独立承担的全局模型聚合与参数广播流程，转化为多节点共同参与的链上共识流程。每一轮模型更新和聚合结果都记录在区块链账本中，并通过加密签名和共识验证确保不可篡改和公开透明。这样，即便某个节点或原先的中心服务器遭遇攻击或失效，其他节点仍然可以通过完整的链上记录恢复全局模型状态，从根本上消除了单点信任风险，并极大增强了联邦学习的抗毁性与安全性。

（2）链上审计增强责任透明[13]。BFL 依托区块链的可追溯与不可篡改特性，为每个联邦训练参与方的行为生成全流程审计凭证。每轮训练提交的本地模型摘要、贡献参数、训练时长、数据标签分布等信息都会固化上链，并绑定参与方的唯一标识。这样，无论是数据质量、训练贡献，还是异常情况的发生时间和责任

方，都可以在链上形成透明完整的证据链条。多方责任的界定与认定从传统的“事后追溯”变为“事中共识”，极大降低了争议解决的成本与难度。

（3）基于贡献大小的动态激励机制[14]。BFL 框架创新性的把智能合约与激励机制融合起来，形成了透明可追溯的贡献评估体系。在实际落实阶段，系统不但要监测各节点上传模型的性能提升幅度，还需综合衡量其提供的数据规模、有效训练时长以及模型稳定性等指标。此框架借助智能合约即时结算贡献值，并把贡献凭证和收益分配直接关联，既保证了高贡献节点拿到超额报偿，又抑制了搭便车的行为，所有评估规则都以代码形式固定于区块链中，借助分布式节点的协同监督，从根本逻辑上重塑了联邦学习生态的信任基础。

区块链联邦学习的现有突破为构建可信分布式学习系统奠定了基石，但其技术演进仍需深度融合传统联邦学习的核心优化方向。最新研究表明，通信效率优化、隐私保护强度提升与公平性评估机制的协同增强，是推动该领域向生产级系统演进的重要路径之一[15]。例如通信效率方面，梯度稀疏化与量化传输技术可显著降低链上通信负载；隐私保护方面，结合差分隐私与轻量级同态加密能构建多级隐私防护体系；而公平性方面，基于 Shapley 值与余弦相似度的动态贡献评估模型则可增强激励公平性。这些技术的有机融合，使得区块链联邦学习在保障可信性的同时，进一步实现工业级系统的需求。

1.2 研究现状与工作挑战

区块链联邦学习作为分布式机器学习与分布式账本技术的交叉领域，其研究进展主要包括通信效率优化、隐私保护增强与公平性评估等关键方向。学术界通过算法创新、协议重构和机制设计，逐步突破链上存储瓶颈、数据隐私风险与参与方激励失衡等核心挑战。本文将从通信高效、隐私增强、公平性三个维度系统梳理技术发展脉络。

1.2.1 通信高效的联邦学习

区块链联邦学习中的通信效率优化主要依赖梯度压缩、异步聚合与模型蒸馏三类技术。相较于异步聚合可能引发的收敛稳定性问题以及模型蒸馏的精度损失挑战，梯度压缩技术因能直接减少链上传输数据量且保持模型性能，成为本文采用的优化方案。现有的梯度压缩研究在梯度量化、稀疏化与编码优化三个子方向取得系列突破，为降低区块链通信负载提供了关键技术支撑。

在梯度量化方向，研究者通过降低梯度数值精度实现通信开销压缩。早期奠基性工作 QSGD 是 Alistarh 等人提出的一种基于梯度量化与编码的通信高效型并

行 SGD 方案，该机制通过灵活调节传输比特数来平衡带宽消耗与收敛速度，在 16 块 GPU 上训练 ResNet-152 网络时，相比全精度训练速度提升了 1.8 倍，显著加速了 ImageNet 上的深度学习训练过程[16]。Tang 等人提出了 1-bit Adam，一种结合误差补偿与分阶段压缩的高效梯度量化方法，在最大 256 块 GPU 的训练实验中，相比标准 Adam，BERT 大规模预训练的吞吐量提高最多 3.3 倍，SQuAD 微调的吞吐量提升 2.9 倍，同时保持与未压缩 Adam 相同的收敛速度[17]。Wang 等人提出了 AQ-SGD，一种针对流水线并行训练的激活量化压缩算法，在慢速网络环境下，通过压缩激活变化，在无损模型质量的前提下，将语言模型微调的端到端速度提升最高 4.3 倍，并结合最优梯度压缩，实现最高 4.9 倍整体加速[18]。

梯度稀疏化通过筛选重要梯度进行传输，显著降低链上存储需求。Li 等人[19]提出了 GGS，一种用于自适应优化器的通用梯度稀疏化框架，通过梯度校正和局部梯度批量归一化更新，在 99.9%梯度稀疏化下依然保持了极高的模型准确率，有效降低联邦学习中的通信开销。Tang 等人[20]提出了 GossipFL，一种去中心化的联邦学习框架，通过稀疏化对等通信与自适应 gossip 矩阵生成算法，有效降低通信流量 38.5%和通信时间 49.8%，同时保持了模型收敛性能与相对精度。Wang 等人[21]提出了一种自适应稀疏化的方差约减联邦学习算法，结合常规稀疏化与自适应筛选高信息量梯度元素的方法，在保证线性收敛率的同时，将通信开销相比现有稀疏化方法至少降低 60%。

编码优化领域的研究聚焦于提升压缩后梯度的传输效率。Wen 等人[22]提出了 TernGrad，一种三元量化的梯度压缩方法，利用逐层量化与梯度裁剪减少分布式深度学习中的通信开销，在 AlexNet 上实现无精度损失甚至精度提升，在 GoogLeNet 上的准确率损失控制在 2%以内，并有效提升了深度网络训练速度。Jiang 等人[23]提出了 SKCompress，一种结合分位数草图、MinMaxSketch、Huffman 编码和 delta-binary 编码的通用梯度压缩框架，在腾讯真实集群实验中，SKCompress 比现有方法快 12 倍，并有效降低了分布式机器学习中稀疏和非均匀梯度的传输开销。Campbell 等人[24]提出了 Malcom-PSGD，一种结合ℓ1 正则化、矢量源编码和抖动量化的去中心化近端 SGD 算法，在非凸分布式学习场景下实现通信成本降低约 75%，并且理论分析与实验均验证了其高效性与收敛性。

1.2.2 隐私增强的联邦学习

区块链联邦学习的隐私保护机制主要包括同态加密、安全聚合与零知识证明三类方法。其中同态加密因支持密文状态下的梯度计算与链上验证，成为本文采

用的核心隐私增强方案。现有研究在半同态加密、部分同态加密和全同态加密取得显著突破，为平衡隐私保护强度与计算效率提供了技术保障。

半同态加密方案在提升计算效率和安全性方面发挥了关键作用。Rivest 等人[25]提出的 RSA 方案，基于大整数因子分解问题，支持加法同态运算，为后续公钥加密系统奠定了基础。Goldwasser 等人[26]设计的 GM 方案，首次引入二次剩余问题，支持二进制加法同态计算，提高了安全性。Paillier[27]提出的 Paillier 方案，基于合数阶剩余类群，支持无限次加法运算，同时具有较高的计算效率，是众多现代密码协议的核心组件。

部分同态加密研究聚焦于优化计算效率与密文紧凑性。Sander 等人[28]提出的 SYY 方案，支持多项 AND 操作及一个 OR/NOT 操作，使其能够在 NC1 电路中进行安全计算。Boneh 等人[29]的 BGN 方案，支持任意次加法和一次乘法，并基于子群判定问题保障安全性。Ishai 等人[30]设计的 IP 方案扩展了 SYY 方案，采用二进制决策图评估分支程序，在提升计算能力的同时增强了灵活性。

全同态加密领域的研究聚焦于提升实用化性能。Gentry 提出的 Gentry 方案[31]，是首个可行的 FHE 方案，基于理想格构造，并引入了自举技术，实现了对密文的无限次运算。Brakerski 等人[32]提出的 BV 方案，利用学习同余问题，提出了去自举方法，降低了计算成本。Dijk 等人[33]的 DGHV 方案，基于整数同余问题，采用了比特级加密结构，在保持安全性的同时简化了计算过程。

1.2.3 公平性联邦学习

公平性评估机制主要包含贡献度量、激励分配与可信审计三类方法，其中 Shapley 值与余弦相似度的融合应用成为本文重点研究方向。现有研究通过算法优化与区块链特性结合，实现了高效、动态的公平性保障体系。

Shapley 值是一种源于博弈论的公平分配方法，常用于评估各参与方在合作游戏中所作贡献的边际值。在联邦学习场景中，Shapley 值被广泛应用于衡量各客户端对全局模型性能的实际影响，从而用于设计激励机制和公平的资源分配方案。在 Shapley 值改进方向，Yang 等人[34]提出了一种基于 Shapley 值与帕累托效率优化的联邦学习激励机制，其中引入第三方监督收益分配，并通过 Shapley 值法确保公平性，当收益未达到帕累托最优时对相关参与者施加惩罚，最终实现纳什均衡，仿真实验验证了该机制的公平性和最优性。Shi 等人[35]提出 FedFAIM，一种基于 Shapley 值的公平感知联邦学习激励机制，该机制通过高效的梯度聚合保证数据质量驱动的聚合公平性，并结合贡献评估和声誉系统实现奖励公平性，实验表明其相比其他非货币 FL 激励机制提供了更强的激励效果。Yang 等人[36]提出了一种基

于增强型 Shapley 值的联邦学习激励机制，该机制结合层次分析法计算多因素权重，以优化收益分配，实验表明该方法相比单因素 Shapley 值法能更公平地反映参与者贡献。Pan 等人[37]提出 FedMDFG，一种结合余弦相似性衡量公平性的多梯度下降联邦学习算法，该方法通过多目标优化计算公平下降方向，并采用低通信成本的行搜索策略优化步长选择，实验结果表明其在收敛性和公平性方面均优于现有 SOTAFL 方法。

余弦相似度是一种常用的相似性度量方法，通过计算两个向量夹角的余弦值，评估它们在方向上的一致性。在联邦学习中，余弦相似度通常用于衡量客户端本地梯度与全局梯度之间的方向一致性，从而反映节点对整体模型优化方向的正向贡献，它应用通过量化梯度方向一致性实现动态激励。Wang 等人[38]提出 FedFV，一种利用余弦相似度检测并缓解梯度冲突的联邦学习算法，该方法通过调整梯度方向和大小，实现公平性提升，并收敛至帕累托平稳解，实验结果表明其在公平性、准确性和效率方面均优于现有方法。Yan 等人[39]提出 AFL-CS，一种基于余弦相似度的异步联邦学习方法，通过分层惩罚项优化模型聚合，提高了在高统计异质性（non-IID）环境下的收敛速度和稳定性，实验表明其性能可超越同步 FL。Wu 等人[40]提出 CKAFL，一种结合余弦相似度的 K-异步联邦学习框架，通过余弦相似性测量延迟梯度的过时性优化服务器聚合，并引入类平衡损失函数缓解客户端非 IID 数据问题，实验结果表明其在非 IID 和 IID 设置下均优于基线方法。Ren 等人[41]提出 CosPer，一种基于余弦相似度的个性化联邦学习框架，该方法通过自适应局部聚合机制优化个性化模型，使聚合权重由全局梯度和局部梯度之间的余弦相似性决定，实验结果表明其数据集上的准确性、公平性和稳健性均优于现有 PFL 方法。

1.2.4 工作挑战

本研究的主要工作挑战体现在公平评估、安全隐私与系统架构三个层面，并且三者相互交织、彼此制约，构成了本文方案设计的整体难点。首先，在公平性评估方面，现有联邦学习中的贡献评估方法往往依赖于明文梯度或原始数据，只有获取真实的训练贡献信息后，才能对各客户端的贡献度进行准确评估。然而，这种做法显然与联邦学习强调的隐私保护目标存在先天冲突。特别是，当系统引入更强的加密与隐私保护机制后，如何在看不到原始梯度和数据的前提下实现准确的贡献评估，成为首要难题。安全增强必然带来隐私提升，但同时也加剧了贡献评估的难度，如何兼顾隐私与公平，本质上是一对难以彻底化解的矛盾，目前

尚无完美解法，本文所做的工作也只能在现有技术约束下寻求最大程度的调和与兼容。

其次，在通信与训练效率层面，引入安全保护和隐私增强势必会对联邦学习的通信开销带来显著提升。特别是当系统采用同态加密、MPC、门限解密等多方安全技术时，数据加密、解密、校验等一系列额外步骤都会拉长训练周期，增加整体开销。因此，如何在保证全局模型训练精度的前提下，通过合理的梯度压缩机制降低带宽与计算消耗，形成精度、隐私与通信效率的动态平衡，是本文面临的第二大挑战。

最后，本文整体框架基于联盟链联邦学习架构，构建了端到端的贡献评估与激励体系。与传统依赖中心协调方的联邦学习不同，本文在系统设计上消除单点信任，转而依赖区块链的共识与可追溯能力实现训练过程的透明可验证。这一设计思路虽然增强了整体可信度，但如何构建一个兼容多种角色、兼顾性能与透明度的高效联盟链框架，并让其无缝承载加密梯度传输、贡献评估记录、历史贡献存证、激励与选举规则执行等复杂功能，也是本文面临的关键工程挑战。

综上所述，本文的技术难点不仅仅是某个单点的创新问题，而是公平评估、安全增强、透明可信系统构建三维一体的系统级挑战，这种多目标约束下的整体设计与技术实现，构成了本文工作的核心价值与最大难点。

1.3 论文的研究内容

本文主要研究的是基于联盟链的联邦学习中的隐私保护、通信效率与贡献评估公平性问题。首先，需要明确联盟链联邦学习的系统架构，分析在该架构下如何兼顾梯度传输效率与客户端贡献评估公平性，同时确保训练过程中全程隐私保护。本文通过对现有的梯度压缩技术、贡献评估方法与联盟链协作机制的系统分析，围绕框架设计、梯度压缩与历史贡献评估三方面展开研究，具体研究内容如下：

（1）针对传统联邦学习依赖单点服务器的信任隐患，提出了一种基于联盟链的联邦学习框架。将联邦训练过程的梯度提交、贡献评估、与节点选举等关键环节全部交由联盟链完成。每轮训练均通过动态选举产生服务节点，验证委员会负责验证与存证，贡献评估委员会负责贡献评估及排名生成。该框架有效消除了中心化信任风险，实现了去中心化、全流程可追溯的可信联邦学习环境。

（2）针对梯度传输带来的高通信开销问题，提出了一种具备表现均衡公平性的动态误差感知梯度压缩机制。该机制基于误差计数器与误差累积器协同工作，实现每个维度的长期未选中次数追踪与未传输梯度信息保存。每轮训练时，客户

端根据计数器生成动态选择概率，优先选取长期未选中的梯度分量，并仅上传这些选中的维度，未选中的残差则继续累积。该机制实现了梯度上传的稀疏化，在有效降低通信开销的同时，保证历史梯度信息的逐轮补偿，从而兼顾了压缩效率、训练精度与表现均衡公平性。

（3）针对贡献评估需要原始梯度的隐私冲突，提出了一种具备贡献评估公平性的基于长期公平的历史贡献评估机制。该机制基于加密梯度与加密余弦相似度，在不解密原始梯度的前提下完成方向贡献评估。评估结果逐轮累积并上链存证，结合指数衰减与角色冷却策略，针对服务节点设计冷却期奖励机制，使其可进入冷却状态，在无需继续上传梯度的情况下，直接获得全局梯度更新的收益。整体机制兼顾贡献透明、历史公平与长期激励，为联邦学习的可持续运行提供激励保障。

1.4 论文的组织架构

本文的组织架构如图 1-1 所示：

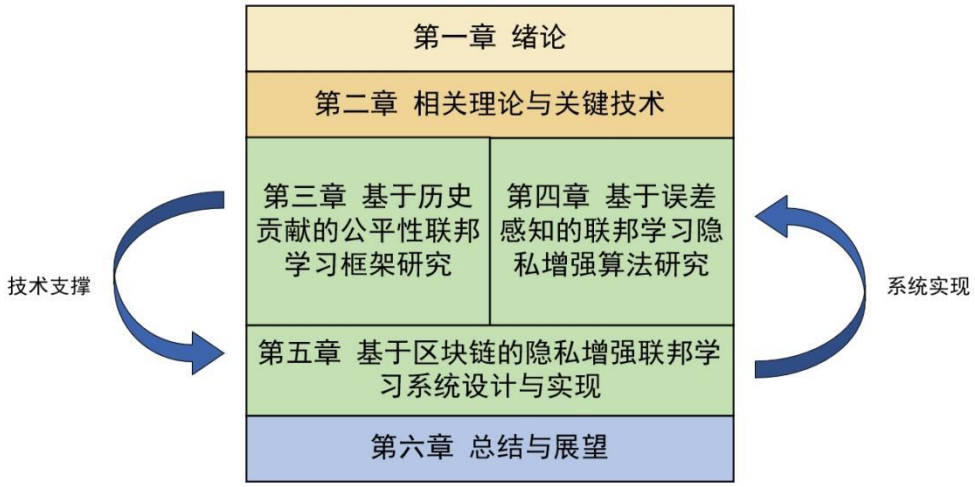


图 1-1 学位论文组织架构

本文分为 6 个章节对上述内容进行研究，具体的文章结构安排如下：

第 1 章：绪论。本章对比传统集中式学习，引出现有联邦学习与区块链技术的融合发展趋向，着重分析了现有的联邦学习中内通信效率优化、隐私保护机制以及公平性评估三个方面的进展与缺陷，进而明确本文核心研究目标及技术方法的达成途径。

第 2 章：相关理论与关键技术。本章介绍本文框架和算法所需的理论基础知识，阐述了联邦学习的经典算法，区块链的关键机制及 Paillier 加密特性呈现，同时针对通信系统优化需求，针对梯度压缩主要方法及其优劣势开展了梳理。

第 3 章：基于历史贡献的公平性联邦学习框架研究。本章提出基于联盟链架构的协作框架，构建包含领导人节点、委员会节点的多方合作机制，同时引入历史贡献评估算法，借助区块链将各节点的累积贡献值存储起来，保障协作流程的透明度与成果认定的可靠度。

第 4 章：基于误差感知的联邦学习隐私增强算法研究。本章聚焦于梯度压缩技术的优化，提出动态误差感知机制，运用 Rand-k 压缩提高通信效率，同时采用 Paillier 加密架构隐私保护的双重屏障，在保障模型收敛趋向的同时增强长期训练的信息可信度。

第 5 章：基于区块链的隐私增强联邦学习系统设计与实现。本章实现了区块链联邦学习系统设计，系统涵盖了模型训练管理、贡献评估管理、用户管理、日志信息管理等关键功能模块，以提高系统的可操作性与管理便利性。

第 6 章：总结与展望。本章对各模块构建思路予以归纳，归纳此次研究在技术上的创新要点，并针对现有方案的局限性，对未来通信优化、隐私计算以及贡献评估机制等方面进行展望。

第二章 相关理论与关键技术

2.1 引言

本章首先对联邦学习的技术架构做系统梳理，阐释其典型训练流程与经典算法 FedAvg 的运行原理，并对横向联邦学习、纵向联邦学习、迁移联邦学习这三种不同模式展开对比分析。就框架构建而言，着重剖析区块链的核心组件与 PBFT 共识机制在分布式场景里的运行逻辑和优势特性。鉴于隐私保护需求，探讨同态加密技术原理，介绍 Paillier 加密方案的加法同态特性及其在参数安全交互中的优势。本章通过基础理论的梳理与比较，为后续系统的优化及实验分析提供理论方面的支撑。

2.2 联邦学习

联邦学习通过客户端与中心服务器的两级架构实现多方协作建模。客户端作为数据持有方，在私有数据集上执行模型训练与梯度计算，其核心功能包括本地迭代优化和参数更新生成。中心服务器负责全局模型的初始化分发和多源参数聚合。典型训练流程中，中心服务器将初始参数广播至客户端，客户端基于本地数据完成前向传播与反向梯度计算后，将更新后的参数加密回传，服务器通过加权平均策略生成新一代全局模型。该架构通过参数协作而非数据共享实现知识融合，有效规避原始数据传输带来的隐私风险。经典横向联邦学习流程如图 2-1 所示。

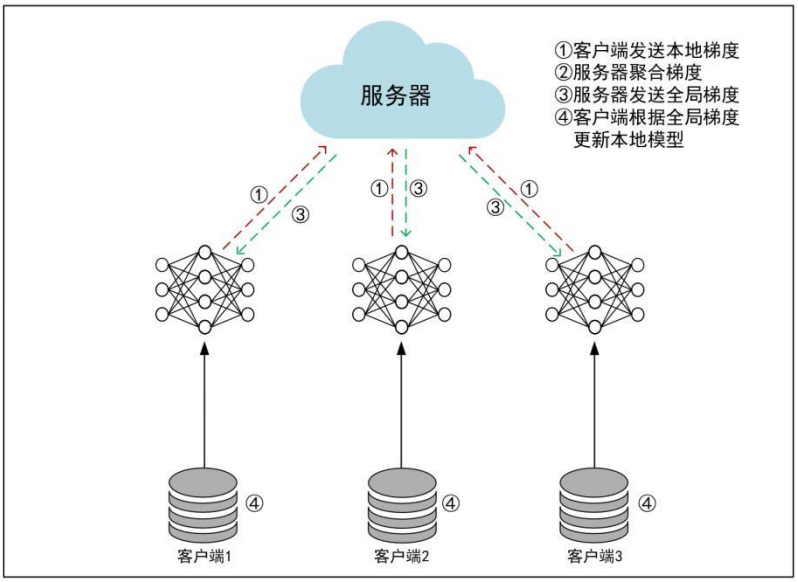


图 2-1 经典横向联邦学习流程

2.2.1 联邦学习的平均算法

联邦平均算法（Federated Averaging, FedAvg）是联邦学习的核心基础算法[42]，其通过加权聚合本地模型更新的方式实现全局优化，既保留了分布式计算的效率，又兼顾了数据隐私保护。该算法的核心思想在于：通过参数平均化而非数据集中化，将分散在多个客户端的数据价值提炼到全局模型中，同时避免原始数据离开本地设备。算法 2-1 描述了 FedAvg 的具体流程。

算法 2-1 FedAvg 算法

1 **输入**：客户端数量 K，第 i 个客户端的本地数据集 D_i，全局数据总量 $N=\sum_{i=1}^{K} n_i$，

最大训练轮数 T，每轮本地训练的迭代次数 E，学习率 η

2 **输出**：最终全局模型参数 w_T

3 服务器初始化全局模型参数 w_0，并分发至所有客户端

4 本地模型更新

for 每个客户端 i **do**

设本地参数初始化为 $w_i^t \leftarrow w^t$

for 迭代 E 轮 **do**

更新模型参数 $w_i^{t+1} \leftarrow w_i^t - \eta \nabla F_i(w_i^t)$

end for

end for

5 每个客户端上传本地训练后的模型参数 w_i^t 至服务器

6 服务器对所有客户端上传的模型参数进行加权平均 $w^{t+1} \leftarrow \sum_{i=1}^{K} \frac{n_i}{N} w_i^t$

7 服务器将更新后的全局模型 w_i^{t+1} 发送至所有客户端

8 **if** 训练收敛或达到最大训练轮数 T **then**

终止训练

else

返回步骤 2 继续训练

end if

9 **返回**：最终全局模型参数 w_T

2.2.2 联邦学习的分类

联邦学习根据数据分布与协作模式的不同，衍生出横向联邦学习、纵向联邦学习和迁移联邦学习三大分支，分别解决不同场景下的隐私协作问题。联邦学习的分类如图 2-2 所示。

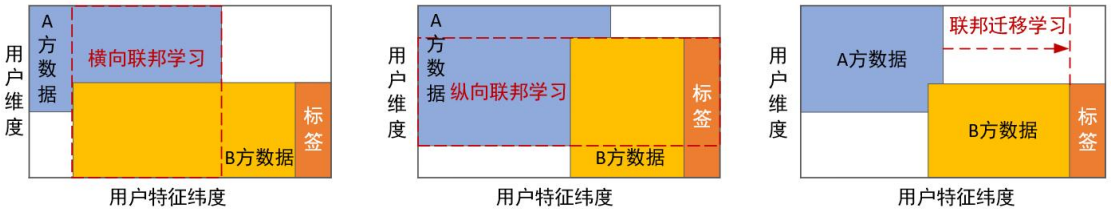


图 2-2 联邦学习的分类

（1）横向联邦学习

当用户维度不重叠但特征维度对齐时适用。例如 A 银行与 B 银行的客户群体完全独立，但双方数据均包含收入水平、信用记录、负债率等相同特征字段。在此场景下，技术实现需解决样本分布异构性带来的模型偏差问题[43]。典型方法包括动态权重分配策略，其中数据量较大的客户端在参数聚合阶段获得更高权重，而本地梯度方差较小的客户端则被赋予更强的模型贡献置信度。

（2）纵向联邦学习

适用于用户维度重叠而特征维度互补的场景。例如 A 医院与 B 保险公司共享同一城市的居民健康数据，居民诊疗记录归属于 A 方持有，保险理赔数据归 B 方持有。核心技术挑战聚焦于跨机构特征空间融合，须通过隐私集合求交界定出共有用户子集，采用安全多方计算达成纵向特征的拼接操作。在此过程中，疾病特征梯度仅由 A 方给出，保险特征梯度的贡献仅来自 B 方，通过同态加密促成非重叠特征达成绝对隔离状态[44]。

（3）迁移联邦学习

当用户维度与特征维度均未出现重叠时采用。例如 A 国的电商平台与 B 国的社交媒体平台，A 方数据是由东南亚用户购买记录组成，B 方的数据反映出欧洲用户的点赞行为情形，两者在用户群体、特征空间上皆不存在重叠情形。该场景需构建起跨域知识迁移机制，借助特征空间映射把 A 方消费模式知识迁移至 B 方社交推荐系统，主要采用的方法有生成共享的隐匿空间，借助对抗训练化解文化差异造成的分布偏差，另外凭借元学习框架提升模型于新区域冷启动方面的能力[45]。

2.3 区块链

区块链技术通过融合密码学原理、分布式系统理论与共识协议构建去中心化信任基础设施，其核心技术体系由三大支柱构成：

（1）链式数据结构[46]

采用哈希指针将交易数据按时间顺序串联为不可逆的链式序列。每个区块包含前序区块的密码学哈希值，形成数据完整性保护机制——任何局部篡改将引发后续所有区块哈希值失配，使得恶意修改可被网络节点快速检测[47]。

（2）非对称加密体系[48]

基于公钥密码学实现网络参与方的身份认证与交易授权。每个节点持有独一无二的密钥对：公开密钥用于验证身份与交易合法性，私有密钥用于生成数字签名。该机制确保交易发起者身份可验证且不可抵赖，同时防止未授权节点伪装参与共识过程[49]。

（3）分布式共识机制[50]

通过数学协议协调去中心化节点的状态同步，解决分布式环境下的数据一致性问题。其中拜占庭容错类共识能够在部分节点故障或作恶时，仍保证诚实节点间达成全局一致性。此类算法通过多阶段消息广播与验证机制，使节点间无需预置信任关系即可协作，为跨域联邦学习架构提供了去中心化协作基础[51]。

2.3.1 PBFT 共识

PBFT（Practical Byzantine Fault Tolerance）作为区块链和分布式系统中普遍应用的共识算法，旨在解决拜占庭将军问题[52]。该算法有能力保证在不可靠与恶意节点的情况下，分布式系统里绝大多数节点依旧能够达成一致，维持系统的正常运转秩序，PBFT 的主要特质是容忍最多三分之一的节点出现故障或恶意的状况，并借助投票机制保障数据一致性。

PBFT 的流程主要包括以下几个步骤：

（1）节点角色分配

PBFT 网络里的节点可划分为三类：主节点、副节点和客户端。主节点承担提出新的区块或事务请求的职责，把提案广播给副节点；副节点负责验证主节点的提案，也参与围绕共识开展的决策。客户端是网络的外部用户，向区块链系统提交交易请求。

（2）请求阶段

客户端把交易请求发送给主节点，该请求含有交易内容及客户端数字签名，来佐证交易的合法性。主节点完成交易格式与签名检查后，进入到下一阶段工作。

（3）预准备阶段

主节点为交易分配唯一编号，并向所有副节点广播预准备消息，包含交易请求及主节点的签名。副节点接收到交易后，查证交易是否合法及主节点签名真伪，若验证达到通过标准，则进入准备阶段。

（4）准备阶段

当客户端向区块链网络提交交易请求时，主节点首先将交易请求作为提案发出。主节点对所有副节点广播这个提案，每个副节点拿到提案的时刻，会针对提案进行核验。如果副节点认为提案合法，它会向主节点发送一个准备消息，表示其已准备好接受这个提案。

（5）承诺阶段

当主节点收到相当数量的准备消息，即说明提案获得了多数节点的拥护认同。主节点会往网络传送一个承诺消息，副节点在接收到主节点的承诺消息之后，会核查所有准备消息是否一致，如果一致副节点会向主节点发送一个赞同承诺的消息，表明其认定该提案。

（6）回复阶段

当一个副节点收到来自三分之二以上节点递送的承诺消息时，说明该提案已得到了充足的支持，并且已经历经了共识流程，在此阶段副节点会回复这个事务，并把该事务录入到区块链，全部节点都将更新各自状态，从而反映该提案的执行成效。PBFT 共识流程如图 2-3 所示。

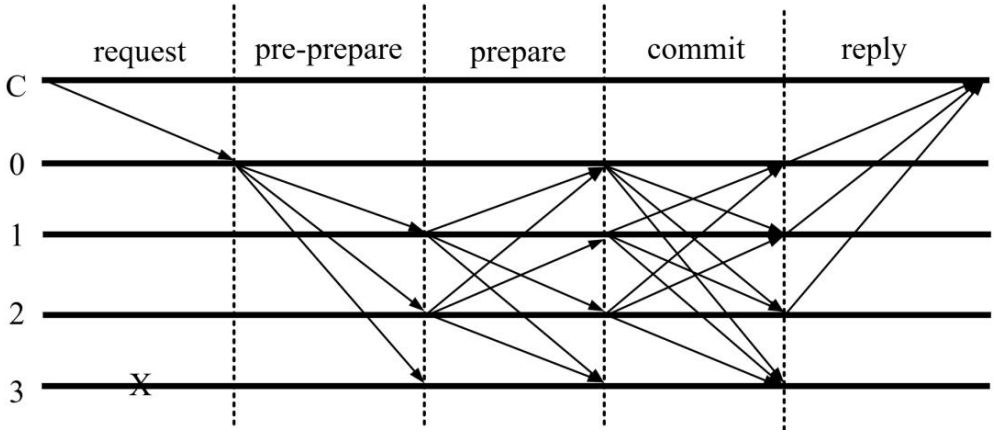


图 2-3 PBFT 共识流程

2.4 同态加密

同态加密技术可追溯至 1978 年 Rivest 等人提出的“隐私同态”概念[53]，其定义为一种允许直接对加密数据执行运算的密码学技术，使得密文计算结果解密后等

价于明文直接计算的结果。该技术通过公钥加密和私钥解密的分离机制，确保数据全程以密文形式参与计算任务，从而在隐私保护与数据可用性之间建立平衡[54]。

2.4.1 同态加密的数学定义

在抽象代数中，同态描述了代数结构之间的映射关系，要求映射在运算下保持不变[55]。设$(G,\oplus)$和$(H,\otimes)$是两个群，映射$f:G\to H$称为群同态，当且仅当对任意$a,b\in G$满足：

$$f(a\oplus b)=f(a)\otimes f(b) \tag{2-1}$$

若进一步考虑环结构同态$(R,+,\cdot)$，则环同态需同时满足：

$$f(a+b)=f(a)+f(b),\quad f(a\cdot b)=f(a)\cdot f(b) \tag{2-2}$$

这一概念在密码学中得到广泛应用，尤其在同态加密中发挥了关键作用。同态加密定义为五元组$(M,C,KeyGen,Enc,Dec)$，其中明文空间M和密文空间C构成代数结构，且对任意明文$m_1,m_2\in M$和操作$\circ\in\{+,\times\}$，存在密文运算$\odot$使得：

$$\mathrm{Dec}\left(\mathrm{Enc}(m_1)\odot\mathrm{Enc}(m_2)\right)=m_1\circ m_2 \tag{2-3}$$

基于运算保持性与实际需求的权衡，现代同态加密技术根据运算能力分为半同态加密（PHE）与全同态加密（FHE）。半同态加密通过限制运算类型实现高效计算，例如 Paillier 方案[56]支持加法同态性：

$$\mathrm{Enc}(m_1)\cdot\mathrm{Enc}(m_2)=\mathrm{Enc}(m_1+m_2) \tag{2-4}$$

而 RSA 方案则实现乘法同态性：

$$\mathrm{Enc}(m_1)\cdot\mathrm{Enc}(m_2)=\mathrm{Enc}(m_1\times m_2) \tag{2-5}$$

全同态加密通过环同态结构突破单一运算限制，以 BGV 方案[57]为例，其支持密文加法与乘法的任意组合：

$$\begin{cases}\mathrm{Enc}(m_1)\oplus\mathrm{Enc}(m_2)=\mathrm{Enc}(m_1+m_2)\\ \mathrm{Enc}(m_1)\otimes\mathrm{Enc}(m_2)=\mathrm{Enc}(m_1\times m_2)\end{cases}, \tag{2-6}$$

总之，同态加密的核心在于构造满足特定代数结构的加密函数，使得加密与计算过程具有可交换性，即加密后的数据在保持保密性的同时仍可进行加法、乘法等操作，从而实现对密文空间的有效运算支持。

2.4.2 Paillier 加密方案

Paillier 加密方案是 Pascal Paillier 于 1999 年提出的公钥加密系统，具有加法同态性，能够直接对加密数据执行加法运算而无需解密。该方案在隐私保护计算和安全多方计算中广泛应用。以下详细介绍其密钥生成、加密解密过程及同态性实

现。Paillier 密钥生成算法、加密算法和解密算法如算法 2-1、算法 2-2、算法 2-3 所示。

算法 2-1 Paillier 密钥生成

输入：安全参数 λ

输出：公钥 (N,g)，私钥 (λ,μ)

选择大素数 p 和 q，满足 $\gcd\left(pq,(p-1)(q-1)\right)=1$

计算 $N \leftarrow pq$

$\lambda \leftarrow \mathrm{lcm}(p-1,q-1)$

while 取样生成元 g **do**

  if g 不满足条件 **then**

    重新取样

  end if

end while

计算 $\mu \leftarrow \left(L\left(g^{\lambda} \bmod N^2\right)\right)^{-1} \bmod N$

返回：公钥 (N,g) 和私钥 (λ,μ)

算法 2-2 Paillier 加密

输入：公钥 (N,g)，明文 $m \in \mathbb{Z}_N$

输出：密文 c

取样随机数 $r \in \mathbb{Z}_N^*$

计算密文 $c \leftarrow g^m \cdot r^N \bmod N^2$

返回：密文 c

算法 2-3 Paillier 解密

输入：私钥 (λ,μ)，密文 c

输出：明文 m

计算辅助值 $x \leftarrow c^{\lambda} \bmod N^2$

应用辅助函数 $L(x) \leftarrow \dfrac{x-1}{N}$

恢复明文 $m \leftarrow L(x) \cdot \mu \bmod N$

返回：明文 m

2.4.3 正确性证明

在证明 Paillier 加密方案的正确性前，本文首先给出几个相关的数学定理，这

些定理用于支撑后续的推导 Paillier 加密的正确性证明：

定理 1（指数展开近似） 对任意整数 x 和足够小的 n，有：

$$(1+n)^x \equiv (1+nx) \bmod n^2 \tag{2-7}$$

定理 2（卡迈克尔定理） 对于素数 p, q 及其乘积 $N = pq$，卡迈克尔函数定义为：

$$\lambda(N) = lcm(p-1, q-1) \tag{2-8}$$

若 r 为 $\mathbb{Z}_N^*$ 的元素，则有：

$$r^{\lambda(N)} \equiv 1 \bmod N \tag{2-9}$$

在介绍上述定理后，开始进行正确性证明：

根据密文公式：

$$\begin{aligned} c^\lambda \bmod N^2 &= (g^m \cdot r^N)^\lambda \bmod N^2 \\ &= g^{m\lambda} \cdot r^{N\lambda} \bmod N^2 \end{aligned} \tag{2-10}$$

由于 Paillier 方案中的生成元 g 选取为：

$$g = (1+N) \bmod N^2 \tag{2-11}$$

因此：

$$g^{m\lambda} \bmod N^2 = (1+N)^{m\lambda} \bmod N^2 \tag{2-12}$$

由定理 1 可知：

$$(1+N)^{m\lambda} \equiv (1+Nm\lambda) \bmod N^2 \tag{2-13}$$

且由定理 2 可知：

$$c^\lambda \bmod N^2 = (1+Nm\lambda) \bmod N^2 \tag{2-14}$$

Paillier 解密使用辅助函数：

$$L(x) \leftarrow \frac{x-1}{N} \tag{2-15}$$

应用于 $c^\lambda \bmod N^2$：

$$L(c^\lambda \bmod N^2) = m\lambda \bmod N \tag{2-16}$$

同理可知：

$$L(g^\lambda \bmod N^2) = \lambda \bmod N \tag{2-17}$$

因此：

$$\frac{L(c^{\lambda} \bmod N^2)}{L(g^{\lambda} \bmod N^2)} = m \tag{2-18}$$

由此得证。

2.5 梯度压缩

梯度压缩是联邦学习里应对通信瓶颈的关键技术，通过针对高维梯度数据开展智能编码与筛选，在降低传输量的同时保持模型收敛性，其方法可归为两类：梯度量化采用降低数值精度的方式压缩数据规模[58]；梯度稀疏化凭借重要性或随机性仅传输定量梯度[59]。这两种技术通过独立运作或协同开展，结合误差补偿机制，达成了通信效率与模型精度的均衡，尤其适配资源有限的移动设备以及隐私敏感场景[60]。

2.5.1 梯度量化

梯度量化通过数值重映射减少单梯度值的存储位数，主要分为三类策略。线性量化[61]将梯度范围均匀划分至低比特区间，例如将[-1.0,1.0]映射到 8 位整数，适用于均匀分布的梯度；非线性量化[62]则采用对数或幂律函数优先编码小幅度梯度，适配自然语言模型中常见的稀疏长尾分布；动态量化[63]通过实时监测梯度分布调整量化步长，避免固定步长导致的信息截断或饱和。量化技术需结合反量化解码与梯度缩放因子，确保服务器端能近似恢复原始数值分布[64]，在联邦学习的跨设备训练中显著降低带宽压力。

2.5.2 梯度稀疏化

梯度稀疏化通过筛选高贡献度梯度实现通信优化，常见方法包括 Top-k 算法与 Rand-k 算法。Top-k 算法基于梯度张量的全局幅值排序机制，构建了确定性高贡献度梯度筛选框架。其核心在于对全量梯度分量的绝对值进行严格降序排列，精确截取前 k%的显著分量作为稀疏化输出，其余分量强制归零。该筛选过程通过显式排序操作建立稳定的稀疏化准则，确保每一轮迭代中仅保留对参数更新方向具有主导性影响的梯度元素[65]。这种机制在数学上等价于对高维梯度空间进行低维子空间投影，其投影方向由梯度幅值的统计分布特性自适应确定，从而在维持高压缩率的同时，最大程度保留驱动模型收敛的核心信息[66]。此外，确定性筛选策略使得算法对梯度分布的非稳态变化具有鲁棒性，在训练中后期梯度幅值趋于平缓时仍能有效识别相对显著的更新分量[67]。Top-k 算法流程如图 2-4 所示。

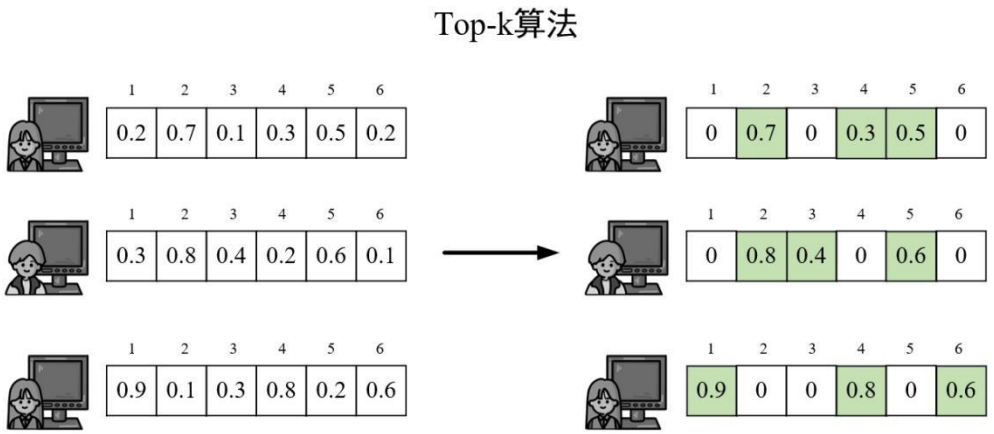


图 2-4 Top-k 算法流程

Rand-k 算法通过构建无偏概率采样架构实现梯度分量的高效随机筛选，其核心在于设计满足期望一致性的稀疏掩码生成规则[68]。算法以预设稀疏率 k%为约束，对梯度张量各维度进行独立或分块关联的随机采样，确保稀疏化后梯度向量的数学期望与原始梯度严格等价。该过程的优势体现在两方面：其一，通过放弃全局排序机制，降低计算复杂度，显著提升高维场景下的处理效率[69]；其二，随机采样引入的隐式正则化效应可增强模型在非凸优化中的探索能力，通过梯度更新方向的适度扰动避免陷入局部极小点。Rand-k 算法流程如图 2-5 所示。

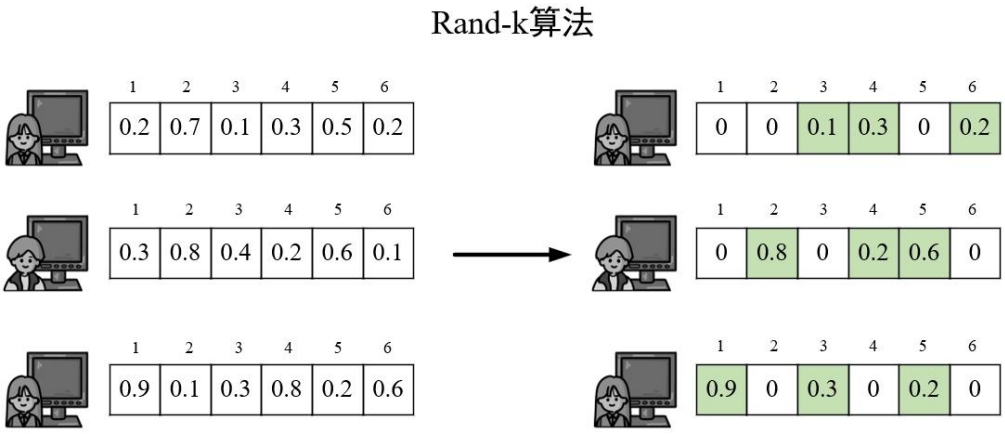


图 2-5 Rand-k 算法流程

总体来看，Top-k 算法依赖显式排序构建确定性稀疏映射，其稀疏化结果具有高度可重现性与选择一致性；而 Rand-k 算法通过构造期望保持的一致性掩码，实现对梯度分量的无偏抽样，其本质为在参数空间中引入受控的随机扰动。两种方法在稀疏策略、计算复杂度与梯度保留方式上呈现出显著差异，分别代表了确定性筛选与概率采样两类主流稀疏化路径。

2.6 本章小结

本章围绕联邦学习的关键技术展开，分析其训练架构、典型算法及不同数据分布下的适用模式，并探讨提升安全性与效率的优化方案。区块链技术通过共识机制增强去中心化协作能力，同态加密确保数据在计算过程中的隐私保护，梯度压缩则减少通信开销，提高联邦学习的训练效率。这些技术为后续系统实现与优化提供了理论支撑。

第三章 基于历史贡献的公平性联邦学习框架研究

3.1 引言

在区块链架构中，公链、私链和联盟链各具特点，但针对联邦学习的实际需求，需全面考量去中心化程度、隐私保护和计算效率等关键因素。公链虽然具备高度去中心化特性，但因所有交易需全网达成共识，其计算成本高、吞吐量低，难以支撑高频的模型更新，同时公链的完全透明性也不利于保护联邦学习过程中涉及的私有数据和模型参数。私链虽然提升了计算效率，并能额外给予更严格的访问控制，但其中心化管理模式与联邦学习的多方协作原则并不匹配。单一机构的治理方式使系统的可信性依赖于特定主体，难以确保数据共享和模型聚合环节的透明性与可追溯性。联盟链把公链与私链的优势相结合，可在保障去中心化的同时提高计算效率与隐私保护能力。其许可准入机制保障参与方身份可信，灵活的共识机制提升计算效率，同时能够支持加密存储和访问控制，增强数据安全性。因此在联邦学习的场景中，联盟链可提供更为合理的技术架构，既保证系统协作时的公平性，又兼顾到隐私保护与计算性能。

在联邦学习场景下，联盟链的引入不仅解决了单点信任问题，还为系统公平性的提升提供了可能。公平性是联邦学习长期可持续运行的关键因素，现有研究围绕三类核心公平目标展开：（1）表现均衡公平性，即确保全局模型在各客户端上的性能一致性，常用标准差[70]、基尼系数[71]、Jain 公平指数[72]等指标衡量；（2）贡献评估公平性，强调基于资源条件[73]、效用影响[74]或验证精度[75]客观评估各客户端的贡献水平，并形成透明的利益分配机制；（3）模型公平性，关注全局模型在敏感群体上的公平表现，兼顾消除群体歧视与提升整体泛化公平[76]。上述公平目标从过程公平与结果公平两个维度，构建了多层次的联邦学习公平性框架，为后续公平性提升技术提供了清晰的优化方向。

在贡献评估公平性方面，已有研究提出了多种机制，其中夏普利值与余弦相似度是两种广泛采用的贡献衡量方法。夏普利值以合作博弈理论为基础，通过逐轮评估各客户端对全局模型的边际贡献，实现全局贡献归因，具备公平性强、理论完备性高的优势。然而，夏普利值的计算复杂度较高，且依赖全量数据参与计算，与联邦学习数据最小化原则和加密传输机制存在天然冲突。相比之下，余弦相似度通过计算客户端上传梯度与全局梯度的方向一致性，衡量其对模型更新的贡献，仅需使用本轮训练数据即可完成评估，既降低了计算复杂度，又兼容现有的加密技术。

基于此，本文提出基于联盟链的去中心化联邦学习框架 FedTide，融合高效模型训练与去中心化治理，通过动态节点轮选机制实现权力制衡，并结合历史贡献评估机制维护长期公平性。具体而言，FedTide 采用加密余弦相似度作为核心贡献度计算方法，同时结合长期表现均衡性需求，进一步优化贡献评估的时间累积策略，构建适应长期公平性的激励机制。

3.2 基于联盟链的去中心化联邦学习框架

3.2.1 节点角色

为实现动态角色轮换与安全协同学习，FedTide 框架围绕四类核心角色构建多维度协作体系：

1、权威机构：作为系统的信任锚点，负责全局密钥的生成、分发与周期性更新，确保加密体系的可信性与抗攻击能力。

2、领导人节点：作为每轮训练的核心聚合执行者，负责主导梯度聚合流程。接收普通用户加密梯度，验证数据签名合法性后，通过同态加密计算生成全局梯度密文。系统为领导人节点设定冷却轮数 τ_L，即节点在完成一轮聚合任务后可进入冷却期，在此期间无需继续参与训练任务即可获得更新后的全局模型，以此作为对其前期计算支出的激励与反馈。

3、委员会节点：作为系统的去中心化治理核心，委员会负责全局数据的可信验证与协作决策，其职能进一步细分为两类：

（1）验证委员会：对领导人聚合结果进行独立复算验证，确保梯度未被篡改；负责区块链数据打包与上链操作。

（2）贡献评估委员会：基于安全多方计算评估用户贡献，动态选举下一轮领导人及委员会成员，实现角色去中心化轮换。

系统同样为共识委员会节点设定冷却轮数 τ_E，使其在完成治理任务后可短暂退出训练流程，同样享有间歇性奖励机制。为保障角色轮换的公平性与去中心化目标，系统约束 τ_L 不小于 τ_E，确保领导人节点在更多轮次中的轮换机会，防止其长期控制核心聚合过程。

4、普通用户节点：作为每轮联邦训练的执行参与者，普通客户端负责从区块链中同步最新的全局模型参数，并在本地完成模型训练与梯度压缩操作。随后，节点对压缩后的梯度结果进行同态加密和数字签名，并将密文梯度包提交至领导人节点，参与本轮的全局模型更新过程。

3.2.2 方案流程

为便于后续对 FedTide 框架的阐述，本文首先对核心符号体系进行了统一定义，以明确各变量在算法流程中的语义和作用。表 3-1 列出了本小节中使用的主要符号。

表 3-1 主要符号表

符号	描述
T	总训练轮次
t	训练轮次编号
m	参与客户端数
i	客户端编号
η	学习率
d	梯度向量维度
k	压缩后保留的维度数
j	梯度维度索引
Δw_i^t	客户端 i 在第 t 轮的原始梯度
$\mathcal{C}(\cdot)$	梯度压缩算子
$E(\cdot)$	同态加密函数
α	历史贡献衰减因子
L_t	第 t 轮领导人节点集合
E_t	第 t 轮委员会节点集合
τ_L	领导人节点冷却轮数
τ_E	委员会节点冷却轮数
P_i^{t+1}	客户端 i 的下一轮采样概率
$accumulator_i^t$	客户端 i 的误差累积器
$counter_{i,j}^t$	客户端 i 在第 t 轮第 j 维误差计数器
$Similarity_i^t$	客户端 i 的第 t 轮梯度相似度
$TotalContrib_i^t$	客户端 i 的第 t 轮累积贡献值
$Rank_i^t$	客户端 i 的第 t 轮有效贡献排名
$GlobalRank_t$	第 t 轮全局贡献排名列表

在统一关键符号后，本文进一步构建了系统流程图，直观展示各角色的交互关系及其在联邦学习过程中的运行逻辑。系统流程图如图 3-1 所示。

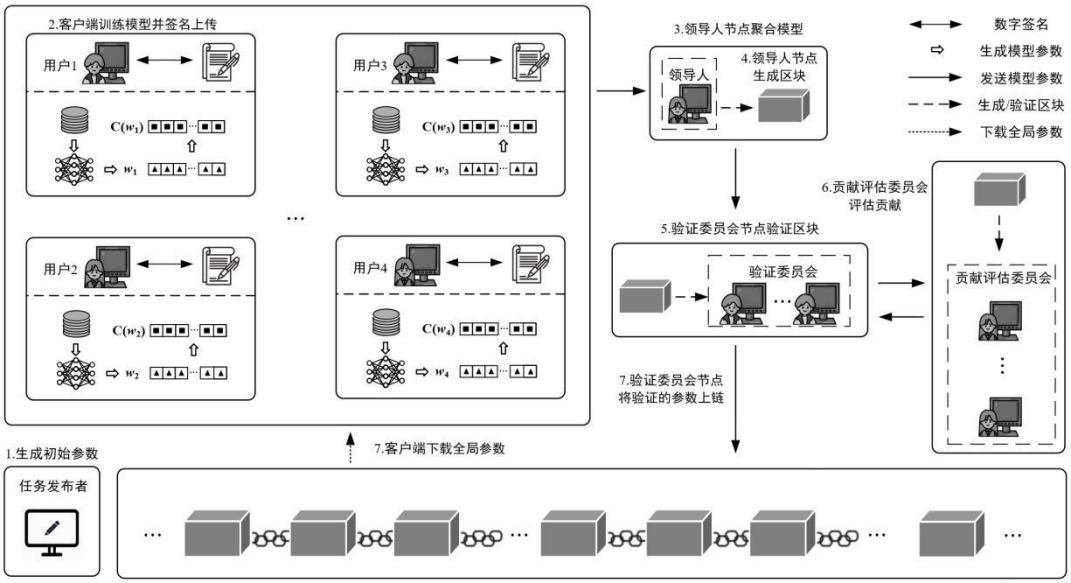


图 3-1 系统流程图

该系统流程主要包含以下步骤：

步骤 1：密钥分发与任务初始化

权威机构 CA 为所有参与节点分发密钥。每个普通用户 i 获得 Paillier 加密公钥 pk_i 和数字签名私钥 sk_i；领导人节点和验证委员会成员仅持有公钥；贡献评估委员会成员额外获得私钥分片。同时，系统初始化全局模型参数 w_0，并确定训练总轮次 T、每轮最大参与客户端数量 m，以及领导人节点的初始集合为 L_0，委员会节点的初始集合为 E_0。

步骤 2：本地训练与压缩

每个客户端 i 在第 t 轮接收来自上一轮区块中的最新全局模型参数 w^{t-1}，并基于本地数据集 D_i 进行训练，计算得到本轮原始梯度 Δw_i^t。随后，客户端基于当前的稀疏掩码 $\mathcal{C}_i^t$ 进行梯度压缩，并结合上一轮残差累积器 $accumulator_i^{t-1}$ 生成补偿梯度。这一步得到压缩后的补偿梯度 $\mathcal{C}(\Delta\tilde{w}_i^t)$。

步骤 3：加密与签名上传

客户端 i 对压缩梯度 $\mathcal{C}(\Delta\tilde{w}_i^t)$ 和对应的平方和进行 Paillier 加密，生成密文 $E(\mathcal{C}(\Delta\tilde{w}_i^t))$ 和 $E(\left\|\mathcal{C}(\Delta\tilde{w}_i^t)\right\|^2)$。同时，客户端生成对数据包的数字签名 σ_i，并附带自身编号 i 和本轮编号 t，通过区块链事务上链，提交至当前轮领导人节点集合 L_t 中的领导人节点。

步骤 4：领导人节点聚合与验签

领导人节点在接收到所有客户端提交的加密梯度后，使用对应公钥 pk_i 验证数字签名 σ_i，确认合法性后，基于同态加法规则对所有 $E(\mathcal{C}(\Delta\tilde{w}_i^t))$ 进行加密聚合，生

成加密全局梯度 $E(\Delta W_{global}^{t})$。此时，领导人节点进入 PBFT 共识的预准备阶段，将聚合结果作为新区块的初步内容，广播至验证委员会成员，启动共识流程。

步骤 5：验证委员会复算校验

验证委员会节点收到领导人节点转发的数据包后，独立复算全局加密梯度 $E(\Delta W_{global}^{t})$，并与领导人节点计算结果比对，确保聚合正确性。此时进入 PBFT 共识的准备阶段，每个验证委员会成员验证新区块的正确性，包括全局模型梯度计算的准确性。如果验证通过，各验证节点向其他委员会成员广播“准备”消息，表示同意该区块内容。

步骤 6：贡献评估与排名计算

贡献评估委员会成员 e 采用 MPC 协议协同生成随机掩码向量 $\{r_k\}_{k=1}^{e}$，并结合加密梯度数据执行安全贡献评估计算，生成每个客户端的第 t 轮贡献分数 $TotalContrib_i^t$。贡献评估委员会进一步汇总各客户端贡献分数，计算本轮全局贡献排名列表 $GlobalRank_t$，并选出下一轮的领导人节点集合 L_{t+1} 和委员会集合 E_{t+1}。贡献评估结果计算完成后，贡献评估委员会将贡献值、历史排名等信息打包，并传输至验证委员会，等待最终确认。同时，系统根据节点的角色履行情况，为上一轮担任领导人节点冷却奖励轮数 τ_L，同时为委员会节点冷却分配奖励轮数 τ_E。在冷却期内，这些节点将暂时退出训练与角色选举流程，但仍可同步全局模型参数，实现计算负担与模型收益的合理分离。

步骤 7：验证委员会打包上链

验证委员会收到贡献评估结果，并进行最终校验。验证通过后，进入 PBFT 共识的提交阶段，将当前轮的加密全局梯度 $E(\Delta W_{global}^{t})$、贡献分数列表 $TotalContrib_i^t$、全局贡献排名 $GlobalRank_t$ 以及下一轮节点选举结果 L_{t+1}、E_{t+1} 打包为新区块 B_t 并上链。

步骤 8：全局模型更新与同步

所有客户端从区块链获取新区块 B_t，解析其中的全局梯度信息并解密得到本轮全局模型更新量，从而同步更新本地模型参数。更新完成后，所有客户端基于最新全局模型和本地数据，启动下一轮训练，循环直至第 T 轮完成。

3.3 基于历史贡献的评估机制

在系统流程明确的基础上，为实现对客户端长期行为的持续监督与激励，本文设计了历史贡献评估机制，用于衡量各参与方在多轮联邦训练中的贡献程度。具体步骤如下：

步骤 1：加密梯度提交与全局聚合

在每轮训练结束后,每个客户端i计算本地训练梯度：

$$\Delta w_i^t = \left[w_{i1}^t, w_{i2}^t, \ldots, w_{id}^t \right] \tag{3-1}$$

为了保护梯度隐私，客户端对每个梯度分量逐一加密：

$$E\left(\Delta w_i^t\right) = \left[E\left(w_{i1}^t\right), E\left(w_{i2}^t\right), \ldots, E\left(w_{id}^t\right) \right] \tag{3-2}$$

同时，为了后续贡献评估中的范数计算，每个客户端本地计算梯度范数平方，并加密上传：

$$E\left(\Delta w_i^{t\,2}\right) = E\left(\sum_{j=1}^{d}\left(w_{ij}^t\right)^2\right) \tag{3-3}$$

领导人节点在收到所有客户端上传的加密梯度后，直接基于 Paillier 同态加法，逐维度对所有客户端加密梯度求和，生成全局梯度：

$$E\left(\Delta W_{\text{global}}^t\right) = \prod_{i=1}^{m} E\left(\Delta w_i^t\right) \tag{3-4}$$

步骤 2：贡献评估委员会生成加密掩码

为了防止点积计算泄露客户端真实梯度信息，贡献评估委员会(共e个成员)分别生成独立随机向量：

$$r_k = \left[r_{k1}, r_{k2}, \ldots, r_{kd} \right] \tag{3-5}$$

每个贡献评估委员会成员对自己的随机掩码逐元素加密：

$$E\left(r_k\right) = \left[E\left(r_{k1}\right), E\left(r_{k2}\right), \ldots, E\left(r_{kd}\right) \right] \tag{3-6}$$

步骤 3：加密余弦相似度分子计算

领导人节点需要计算客户端上传梯度与全局梯度的加密点积：

$$E\left(dot_i^t\right) = \prod_{j=1}^{d} E(w_{ij}^t)^{W_{global,j}^t} \tag{3-7}$$

考虑到 Paillier 加法同态无法直接支持乘法，这里采用加密点积与随机掩码相结合的策略。领导人节点将贡献评估委员会生成的加密随机掩码逐维度叠加到点积结果：

$$E\left(\text{dot}_i^t + R\right) = E\left(\text{dot}_i^t\right) \cdot \prod_{k=1}^{e} E\left(r_k\right) \tag{3-8}$$

其中掩码 R 为：

$$R=\sum_{k=1}^{e} r_k \tag{3-9}$$

贡献评估委员会协同执行门限解密：

$$\mathrm{dot}_i^t + R = Threshold\_Dec\left(E\left(\mathrm{dot}_i^t + R\right)\right) \tag{3-10}$$

解密后，贡献评估委员会去除自身生成的随机掩码：

$$\mathrm{dot}_i^t = \left(\mathrm{dot}_i^t + R\right) - R \tag{3-11}$$

步骤 4：加密余弦相似度分母计算

贡献评估委员会对加密范数平方执行门限解密：

$$\Delta w_i^t = \sqrt{\mathrm{Threshold\_Dec}\left(E\left(\Delta w_i^{t\,2}\right)\right)} \tag{3-12}$$

步骤 5：相似度计算与贡献值更新

贡献评估委员会基于解密后的点积和范数计算相似度：

$$\mathrm{Similarity}_i^t = \frac{\mathrm{dot}_i^t}{\Delta w_i^t} \tag{3-13}$$

每轮的全局梯度是客户端共有的，在这里全局梯度范数可作为共有项省去。

步骤 6：贡献排名生成与上链

每轮贡献排名生成规则：

$$\mathrm{TotalContrib}_i^t = \alpha \cdot \mathrm{TotalContrib}_i^{t-1} + (1-\alpha) \cdot \mathrm{Similarity}_i^t \tag{3-14}$$

$$\mathrm{Rank}_i^t = \begin{cases} 0, & i \in L_t \cup E_t \\ \mathrm{TotalContrib}_i^t, & i \notin L_t \cup E_t \end{cases} \tag{3-15}$$

其中 L_t 为第 t 轮领导人节点集合，E_t 为第 t 轮贡献评估委员会集合。

领导人汇总全体客户端贡献排名：

$$\mathrm{GlobalRank}^t = \left\{\mathrm{Rank}_1^t, \mathrm{Rank}_2^t, \ldots, \mathrm{Rank}_m^t\right\} \tag{3-16}$$

$\mathrm{GlobalRank}^t$ 上链存证，形成可追溯贡献历史。此时可按由高到低的顺序决定下一轮的领导人节点、验证委员会节点和共识评估委员会节点。贡献值评估算法如算法 3-1 所示：

算法 3-1 贡献值评估算法

输入： 当前梯度 Δw_i^t，全局聚合梯度 $\Delta W_{\text{global}}^t$，客户端 i 的历史累积贡献 $\text{TotalContrib}_i^{t-1}$，衰减因子 α，i 的角色状态(是否为领导人/委员会节点)

输出： 当前贡献值 TotalContrib_i^t，有效排名 Rank_i^t

if $t=0$ **then**

$\text{TotalContrib}_i^0 \leftarrow 0$

end if

计算点积 $\text{dot\_product} \leftarrow 0$

for 每个维度 $j \in \{1,\ldots,d\}$ do

$\text{dot\_product} \leftarrow \text{dot\_product} + \Delta w_{i,j}^t \cdot \Delta W_{\text{global},j}^t$

end for

计算 L2 范数 $\text{norm} \leftarrow 0$

for 每个维度 $j \in \{1,\ldots,d\}$ do

$\text{norm} \leftarrow \text{norm} + \left(\Delta w_{i,j}^t\right)^2$

end for

$\text{norm} \leftarrow \sqrt{\text{norm}}$

计算梯度相似度

if $\text{norm} \neq 0$ **then**

$$\text{Similarity}_i^t \leftarrow \frac{\text{dot\_product}}{\text{norm}}$$

else

$\text{Similarity}_i^t \leftarrow 0$

end if

更新累积贡献值 $\text{TotalContrib}_i^t \leftarrow \alpha \cdot \text{TotalContrib}_i^{t-1} + (1-\alpha) \cdot \text{Similarity}_i^t$

生成贡献排名

if i 是领导人节点或委员会节点 **then**

$\text{Rank}_i^t \leftarrow 0$

else

$\text{Rank}_i^t \leftarrow \text{TotalContrib}_i^t$

end if

返回： 当前贡献值 TotalContrib_i^t，有效排名 Rank_i^t

3.4 安全性分析

（1）抗单点攻击

为抵御单点攻击，单点攻击是指某个关键角色（如领导人节点、贡献评估委员会成员）被长久控制或篡改，从而引发系统整体安全的稳定性波动。

本方案运用动态角色分配机制，保障领导人节点与委员会节点定时轮换，杜绝某个节点长期承担聚合、验证和贡献评估等关键工作，同时联盟链的共识机制实现所有决策可追溯、不可篡改，从根源上降低单点攻击的风险。

（2）抗合谋攻击

合谋攻击是指多个恶意客户端联合，操控贡献评估结果、篡改梯度数据，或者有意偏袒某些特定客户端，以获得不公平的奖励。攻击者可凭借相互虚构贡献数据，提升自身在系统里的权重，进而以不合理手段影响全局模型的训练趋向。

为抵御合谋攻击，此方案采用联盟链的存证机制，保证各项的贡献评估结果公开透明且无法篡改，同时本文构建了动态角色轮换机制，使领导人节点以及贡献评估委员会成员依照轮次更替，防范特定节点长久掌控系统，这一综合策略实现了系统的公平性和鲁棒性，切实降低合谋攻击的风险。

（3）抗搭便车攻击

搭便车攻击是指某些客户端在联邦学习阶段不执行本地训练或提交无效梯度，但仍打算获取全局模型更新的奖励。这些恶意节点企图绕开正常的训练流程，依赖其他客户端的付出来提升自己的模型性能，但不给予任何有价值的梯度反馈，搭便车现象不仅对系统的公平性产生影响，还可能使全局模型的训练质量下降，服务器在开展模型聚合的时候，或许也会运用这些无效梯度。

为抵御搭便车攻击，本方案提出了一种基于历史贡献的评估机制，凭借加密余弦相似度去评估客户端的真实贡献度。各个客户端的贡献情况被储存在联盟链上，保障记录的透明性和不可篡改性。就长期贡献偏低的客户端而言，系统会安排此客户端充当服务节点，通过消耗自身算力去支撑全局模型训练并获取对应奖励，从而保证只有真正的贡献者才能享有模型的相关收益。这种基于联盟链的可信激励机制，可以有效应对搭便车攻击，保障联邦学习的公平性与稳定性。

3.5 公平性分析

本章围绕贡献评估公平性和长期公平性展开探讨，分析系统如何保障各节点贡献得到合理估量，并在长期阶段实现维持公平的激励分配。

（1）贡献评估公平性

贡献评估公平性涉及系统是否能精准、公开地衡量各节点的实际贡献，保障

高贡献者获取相应报酬，同时防范搭便车现象。本文采用通过加密余弦相似度得出的归一化贡献值作为核心评估标准，结合联盟链存储贡献评估的综合结果，以实现数据不可篡改，提高透明水平。

为防止贡献评估受到短期数据波动的干扰，系统引入历史贡献的周期累计，让节点的贡献值在多轮训练里逐渐积累，保障评估结果稳定水平提高，同时贡献计算根据梯度变化的相似次序开展，使得贡献评估可公正展现节点对全局模型的实际成果意义。此外系统运用的去中心化评估模式避免了单点操控，进一步巩固了贡献评估的公平性。

（2）长期公平性

长期公平性关心系统在长时间运行阶段的公平性，保证节点的贡献不会因短期起伏而被忽略，同时防止长期积累带来的不公平竞争。本文运用指数衰减策略，让历史贡献随时间逐步缩减，保障近期贡献体现出更大影响力，由此激励节点持续参与以综合历史记录度量贡献。

此外，系统引入冷却奖励机制，容许在前一轮承接计算任务的领导人节点和贡献委员会节点在冷却期内依然能拿到部分收益，以均摊计算资源的消耗，避免长期计算负荷不均衡。但冷却奖励的分配方式经过限定，保证不会对整体激励公平性造成干扰，使所有节点皆能在合理竞争里得到收益。

综上，本文的公平性机制借助历史贡献累积、指数衰减策略以及冷却奖励，在贡献评估和长期收益分配上保障了公平性，维护系统运行的稳定性以及激励机制的可持续发展。

3.6 实验设计与分析

在实验设计之前，需要明确实验的整体设计和实施方案，以确保结果的可靠性和可复现性。实验准备部分涵盖了硬件环境、软件依赖以及数据集的选取和预处理，这些因素共同构建了实验的基础环境，为后续实验的顺利进行提供支持。

3.6.1 实验准备

本实验在配备 Intel Xeon Gold 5218 CPU 和 NVIDIA Tesla V100 GPU 的计算节点上开展，采用 Ubuntu 20.04 作为操作系统，并采用 PyTorch 作为主要深度学习框架。计算节点包含 32 核（16 核/线程双路超线程）、16GB 内存以及 Samsung SSD（446GB）和 Seagate HDD（3.7TB）存储，为大规模数据处理与深度学习计算提供了稳定的硬件支持。

实验环境基于 Python 3.8 搭建，并安装了 PyTorch 及其依赖的 CUDA 11.4，以

实现高效的 GPU 加速计算硬件支持。此外实验依赖于 NumPy、SciPy、稳定版本的 scikit-learn 等核心科学计算库，以达到数值计算的准确性与实验结果的可复现性。硬件配置与软件环境如表 3-2 所示。

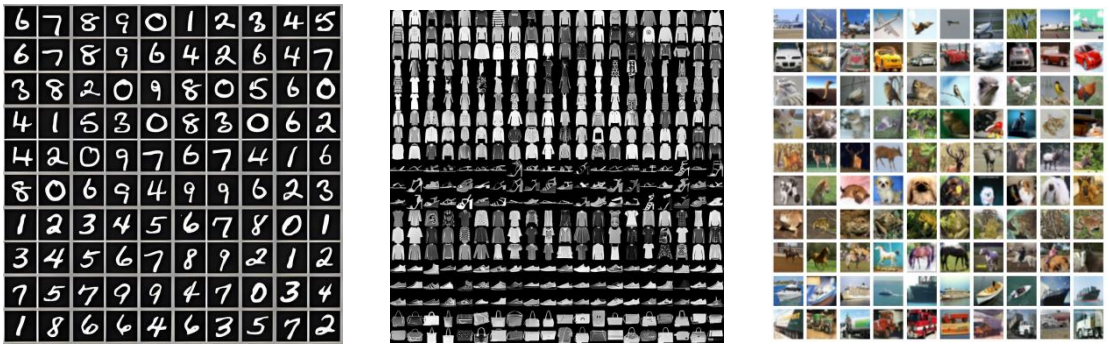
表 3-2 硬件配置与软件环境

说明	参数/版本
操作系统	Ubuntu 20.04
CPU	Intel Xeon Gold 5218 @ 2.30GHz
GPU	Tesla V100-PCIE
Memory	16GB
SSD	4T
CUDA	11.4
Anaconda	3.7
Python	3.8
Pytorch	1.6

本实验选取的数据集包括：MNIST 数据集、Fashion-MNIST 数据集和 CIFAR-10 数据集。MNIST 数据集包含 0 至 9 的手写数字图像，是计算机视觉领域的基准数据集之一。该数据集共包含 70,000 张灰度图像，其中 60,000 张为训练集，10,000 张为测试集。每张图像的分辨率为 28×28 像素，背景为纯黑色，手写数字居中显示。MNIST 因数据格式规范、类别分布均匀，常被用于验证图像分类算法的基本性能。

Fashion-MNIST 数据集包含 10 种服饰类别（如 T 恤、裤子、凉鞋等），总数据量为 70,000 张灰度图像，其中 60,000 张用于训练，10,000 张用于测试。每张图像尺寸为 28×28 像素，数据组织形式与 MNIST 一致，但类别特征复杂度显著提高。该数据集常被用于测试模型对纹理、轮廓等细节特征的提取能力。

CIFAR-10 数据集由 10 类自然物体（如飞机、鸟类、卡车等）的彩色图像组成，总数据量为 60,000 张，包括 50,000 张训练图像和 10,000 张测试图像。每张图像分辨率为 32×32 像素，包含 RGB 三个颜色通道。该数据集因低分辨率与真实场景的多样性，被广泛用于验证模型在复杂特征学习中的鲁棒性。采用的数据集如图 3-2 所示。



(a) MNIST 数据集　　(b) Fashion-MNIST 数据集　　(c) CIFAR-10 数据集

图 3-2 采用的数据集

本章采用 CNN 和 ResNet-18 作为基础模型进行校验。其中，CNN 架构由两层卷积层和一层全连接层组成，可提取局部特征并开展基础的图像分类任务。第一层卷积层用于初步提取边缘和纹理特征，第二层卷积层进一步深入学习更深层的模式呈现，且借助池化层减少计算复杂度，最终特征实施展平操作后输入全连接层做分类预测，该结构所含参数量少，可充当轻量级模型对比实验的基线。

为了评估方法在深度网络中的适应性，本文引入 ResNet-18 开展对比检验。ResNet-18 是一种 18 层深度残差网络，由 17 个卷积层和 1 个全连接层组成，核心结构包含 4 个残差块，每个块由两个 3×3 卷积层组成，并依靠跳跃连接缓解梯度消失问题。网络首层采用 7×7 卷积核进行特征提取，末端利用全局平均池化减少特征维度。该模型参数量约 1,100 万，适用于更复杂的特征学习任务，在深度神经网络训练中展现出较好的稳定性。结合 CNN 和 ResNet-18 开展实验，有助于评估本文方法在不同网络架构下的性能表现。

3.6.2 实验设计

本章围绕提出的基于联盟链的联邦学习框架及贡献度评估算法为核心展开实验分析，核心目标是验证本文方法在不同数据分布环境下的收敛性能、贡献度评估的合理性及系统对恶意节点的鲁棒性。实验针对框架整体流程进行评估，检测其在联邦训练中的稳定性及收敛程度。随后引入独立同分布与非独立同分布两种数据划分方式，以检验算法在不同数据异质性环境下的适应性。鉴于联邦学习的多方协作特点，贡献度评估算法对于贡献评估公平性及长期公平性十分重要，因此，实验进一步对贡献评估方法的有效性展开分析，包括客户端当选领导人节点的分布情况及系统遭受恶意节点干扰时的稳健性。

基于上述目标，本文设计了如下实验：

实验一：针对本文提出的联邦学习优化方法，分析在 MNIST、Fashion-MNIST 和 CIFAR-10 数据集上，独立同分布场景下与 FedAvg、FedBN[77]、SCAFFOLD[78] 算法的模型性能比较。

实验二：针对本文提出的联邦学习优化方法，分析在 MNIST、Fashion-MNIST 和 CIFAR-10 数据集上，非独立同分布场景下与 FedAvg、FedBN、SCAFFOLD 算法的异构数据适应能力比较。

实验三：针对本文提出的贡献度评估算法，分析在 MNIST 数据集上的客户端当选领导人节点的次数，以验证历史贡献对角色选举的影响。

实验四：针对本文提出的贡献度评估算法，分析在 MNIST 数据集上，不同占比的恶意节点对模型性能的影响，评估系统的稳健性及鲁棒性。

3.6.3 实验参数设置

四组实验均基于联邦学习基本框架展开，除特殊说明外，统一设置如下：客户端总数为 30（total_clients=30），每轮随机选取 15 个客户端参与训练（per_round_workers=15），联邦训练总轮次为 100 轮（global_epochs=100），每个客户端本地训练 3 轮（local_epochs=3），批次大小为 64（batch_size=64）。

实验一：采用统一参数配置，构建独立同分布数据环境，验证所提方法在标准数据分布下的模型性能表现。

实验二：参数设置与实验一相同，仅在数据分布上构建非独立同分布环境，以考察所提方法在数据异构场景下的适应能力。

实验三：在统一参数配置下，扩展设置客户端总数为 30 和 40 两种情况，每轮选出 1 个领导人节点、4 个验证委员会节点与 4 个贡献评估委员会节点，训练总轮次为 100 轮。贡献评估参数设定为 0.3，用于分析历史贡献对角色选举机制的影响。

实验四：在统一参数配置下，引入占比为 0%、5%、10%与 20%的恶意客户端（b=0%，5%，10%，20%），通过上传反转梯度（即对本地梯度取负）的方式进行攻击，干扰全局模型聚合过程，评估系统在不同攻击强度下的鲁棒性与稳定性。

3.6.4 实验结果分析

实验一：在 MNIST 数据集下，SCAFFOLD 和本文算法在 30 轮左右达到收敛，最终的准确率结果分别为 96.27%和 97.1%。FedAvg 和 FedBN 收敛速度较慢，大约在 35 轮收敛，最终实现的准确率结果分别为 94.62%和 94.8%。MNIST 数据集 IID 场景下不同算法的性能比较如图 3-3 所示。

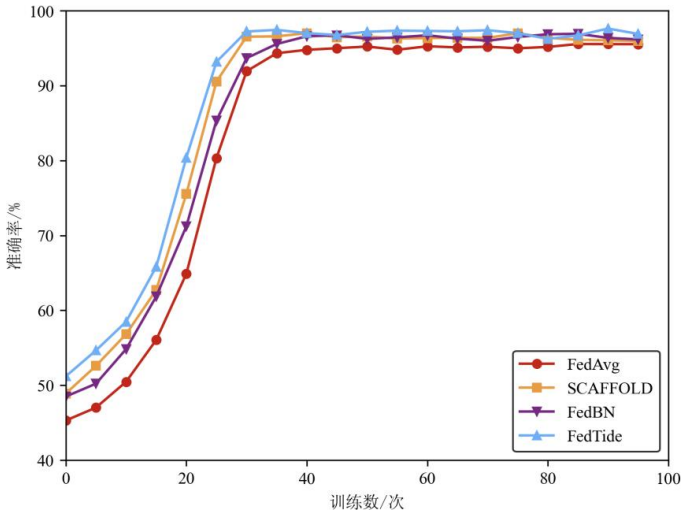


图 3-3 MNIST 数据集 IID 场景下不同算法的性能比较

在 Fashion-MNIST 数据集下，FedBN、SCAFFOLD 和本文算法均在 38 轮左右收敛，最终准确率分别为 88.3%、90.1%和 92.4%。FedAvg 收敛较慢，在 44 轮收敛，最终准确率为 86.2%。Fashion-MNIST 数据集 IID 场景下不同算法的性能比较如图 3-4 所示。

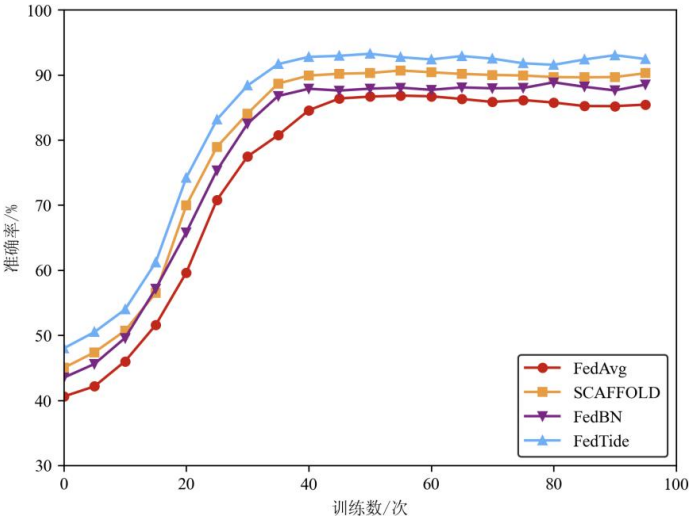


图 3-4 Fashion-MNIST 数据集 IID 场景下不同算法的性能比较

在 CIFAR-10 数据集下，FedAvg 在 54 轮左右收敛，最终准确率为 75.1%。FedBN 和 SCAFFOLD 分别在 50 轮左右收敛，准确率为 78.2%和 80.3%。本文算法收敛最快，在 46 轮完成收敛，最终准确率达到 82.4%。CIFAR-10 数据集 IID 场景下不同算法的性能比较如图 3-5 所示。

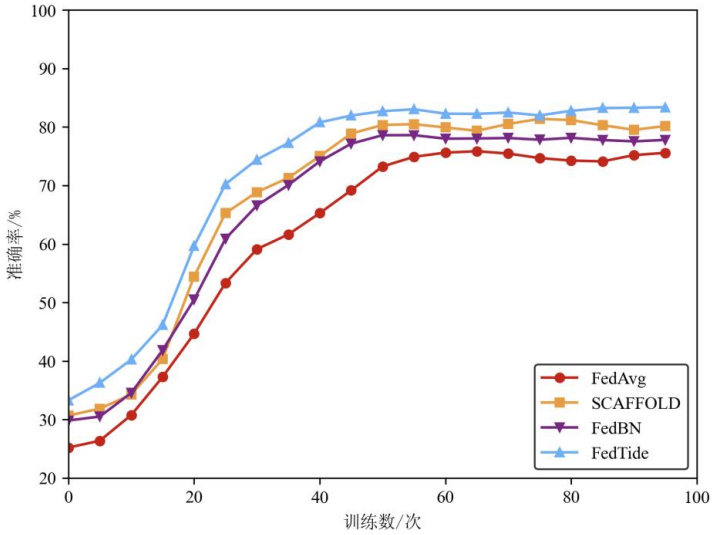


图 3-5 CIFAR-10 数据集 IID 场景下不同算法的性能比较

实验二：在 MNIST 数据集下，FedAvg 因不适应 Non-IID 数据，收敛较慢，在 49 轮收敛，最终准确率仅 76.3%。FedBN 表现较好，在 45 轮收敛，最终准确率为 85.8%。SCAFFOLD 效果最佳，在 40 轮收敛，准确率达到 87.2%。本文算法与 SCAFFOLD 收敛轮次相近，但最终准确率略低，为 86.9%，两者表现基本一致。MNIST 数据集 Non-IID 场景下不同算法的性能比较如图 3-6 所示。

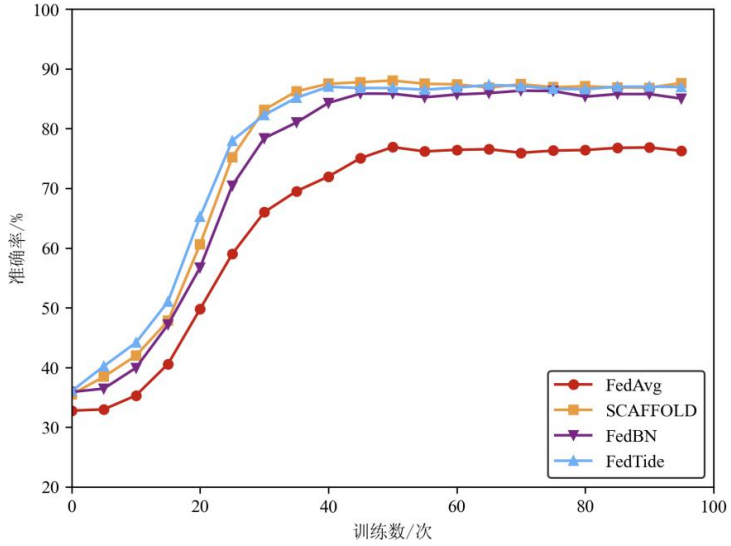


图 3-6 MNIST 数据集 Non-IID 场景下不同算法的性能比较

在 Fashion-MNIST 数据集下，根据图 5，FedAvg 在 62 轮才完成收敛，最终准确率 65.6%，表现较差。其余三种方法性能接近，均在 55 轮左右完成收敛，最终准确率分别为 77.2%、78.5%和 78.7%。Fashion-MNIST 数据集 Non-IID 场景下不同算法的性能比较如图 3-7 所示。

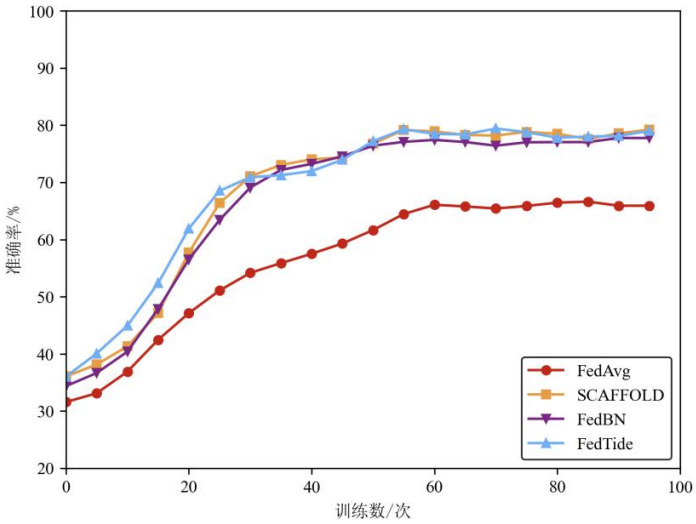


图 3-7 Fashion-MNIST 数据集 Non-IID 场景下不同算法的性能比较

在 CIFAR-10 数据集下，SCAFFOLD 收敛最快，在 60 轮达到稳定，最终准确率 75.0%。FedBN 和本文算法在 65 轮收敛，准确率分别为 72.9%和 74.5%。FedAvg 表现不佳，训练过程中未能有效收敛，在 77 轮后准确率仍仅为 52.1%。CIFAR-10 数据集 Non-IID 场景下不同算法的性能比较如图 3-8 所示。

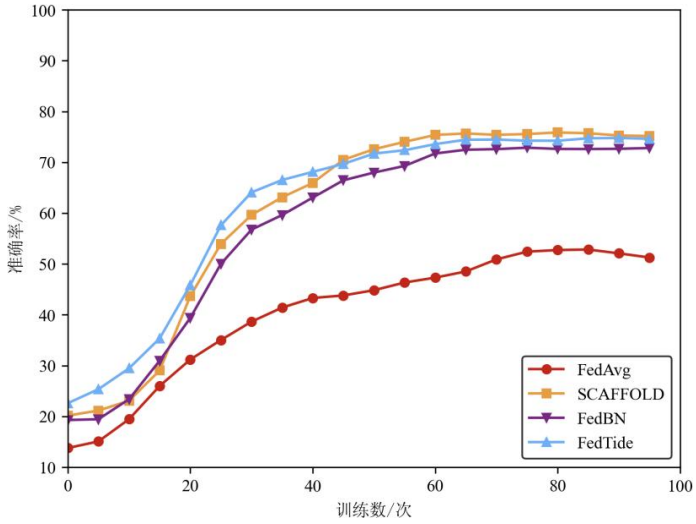


图 3-8 CIFAR-10 数据集 Non-IID 场景下不同算法的性能比较

实验三：在 MNIST 数据集下，客户端总数为 30 时，领导人节点的当选次数整体分布较为均衡。其中，有 18 个客户端当选 3 次，处于主要分布区间；另有 5 个客户端当选 5 次，4 个客户端当选 4 次，3 个客户端当选 2 次，整体结果符合贡献度驱动下的轮换策略，体现出良好的公平性与轮换稳定性。客户端总数为 30 时，领导人节点当选次数统计如图 3-9 所示。

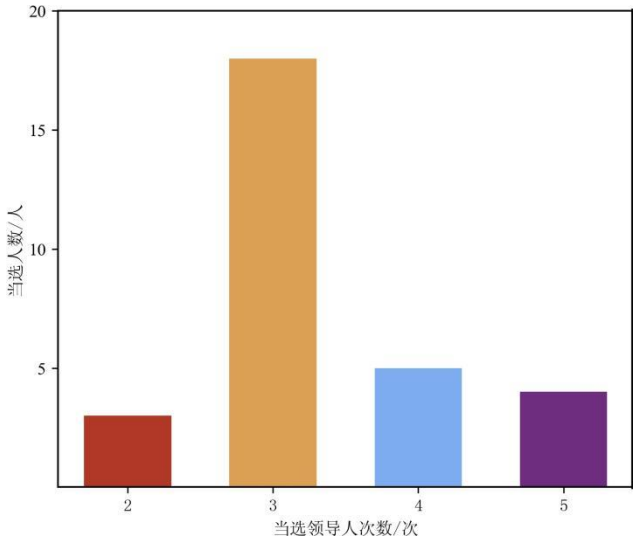


图 3-9 领导人节点当选次数统计（30 个客户端）

当客户端总数增加至 40 时，领导人节点的当选次数相应降低，但分布同样集中。其中，有 15 个客户端当选 2 次，16 个客户端当选 3 次，6 个客户端当选 1 次，4 个客户端当选 4 次，整体结果同样符合预期的角色选举趋势。客户端总数为 40 时，领导人节点当选次数统计如图 3-10 所示。

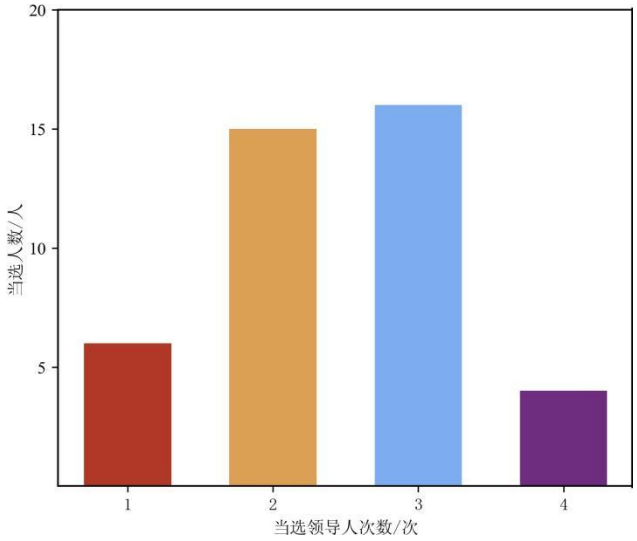


图 3-10 领导人节点当选次数统计（40 个客户端）

实验四：在 MNIST 数据集下，系统在无恶意节点时可正常运行，在约 30 轮内完成收敛，最终准确率达到 97.23%；当恶意节点占比为 5%时，系统依然保持良好的稳定性，尽管收敛速度略有延后至 45 轮，最终准确率为 94.68%。当占比提升至 10%后，系统收敛速度明显减缓，在第 57 轮左右才趋于稳定，最终准确率为 92.28%。当恶意节点占比进一步增加至 20%时，系统训练初期出现准确率波动，

整体收敛过程变得缓慢，最终在第 68 轮左右收敛，准确率为 78.22%，性能虽有所下降，但仍保持基本有效的学习能力，展现出一定程度的鲁棒性。不同恶意节点占比对模型性能影响如图 3-11 所示。

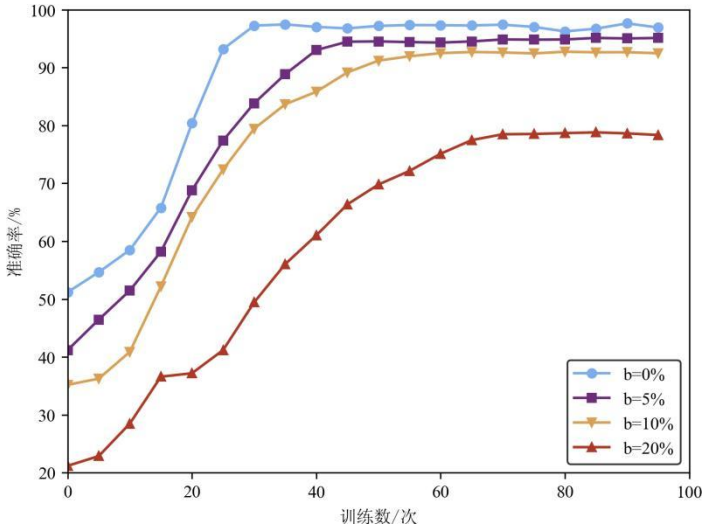


图 3-11 不同恶意节点占比对模型性能的影响

3.7 本章小结

本章围绕提出的基于联盟链的联邦学习优化框架展开设计与验证，通过构建多角色协同机制、引入历史贡献驱动的选举策略及冷却奖励机制，实现了系统的动态轮换与公平激励。同时，通过多组实验验证了该方法在模型性能、公平性与鲁棒性方面的综合优势。

第四章 基于误差感知的联邦学习隐私增强算法研究

4.1 引言

前文已系统阐述 Top-k 与 Rand-k 算法的核心技术特性。尽管 Top-k 算法通过确定性筛选高贡献梯度分量在理论收敛性上具有优势，但其依赖全局排序的操作导致计算复杂度随模型参数量呈超线性增长，且当梯度绝对值分布呈现显著偏态时，筛选结果的客户端间差异性可能引发参数更新方向失衡。相较而言，Rand-k 算法凭借概率采样机制展现出三方面显著优势：

（1）计算效率优势：Rand-k 无需排序操作，计算过程更加高效，尤其适用于参数量庞大的分布式训练场景，在保持压缩效果的同时显著降低了计算开销；

（2）隐私保护增强：采样结果的随机性使得每轮上传的梯度子集不可预测，从而提升了模型更新的随机性与不确定性，有助于增强系统对基于更新模式的反向推理攻击的抵抗能力；

（3）数据异构鲁棒性：在非独立同分布的数据场景下，Rand-k 的多样性采样机制能够有效缓解客户端间梯度方向冲突，提升模型在异构数据环境中的稳定性。

基于上述优势，本文选择以 Rand-k 算法为核心构建梯度压缩框架。为克服朴素 Rand-k 因随机丢弃梯度分量导致的收敛波动问题，进一步引入误差补偿机制，将本轮未选中的梯度残差累积至后续更新中，以渐进修正模型偏差。

4.2 动态误差感知的梯度压缩机制

4.2.1 核心组件定义

定义 1（误差计数器）

每个客户端维护一个维度为 d 的整数向量 $c_i^t \in \mathbb{N}^d$，其元素 $c_{i,j}^t$ 表示客户端 i 在第 t 轮训练时维度 j 的历史未选中累积次数。更新规则由下式严格定义：

$$c_{i,j}^t = I\left(\mathcal{C}_{i,j}^t = 1\right)\cdot 1 + I\left(\mathcal{C}_{i,j}^t = 0\right)\cdot\left(c_{i,j}^{t-1} + 1\right) \tag{4-1}$$

其中 $I(\cdot)$ 为指示函数，$\mathcal{C}_{i,j}^t \in \{0,1\}$ 表示维度选择状态。此设计保证被选中的维度计数器重置为 1（含本轮选中后的增量），未选中的维度计数器递增 1。

定义 2（误差累积器）

误差累积器 $e_i^t \in \mathbb{R}^d$ 是一个动态向量，其更新规则为：

$$e_i^t = \left(\Delta w_i^t + e_i^{t-1}\right) - \mathcal{M}_k\left(\Delta w_i^t + e_i^{t-1}\right) \tag{4-2}$$

其中 $\mathcal{M}_k : \mathbb{R}^d \to \mathbb{R}^d$ 为压缩映射算子，满足 $\mathcal{M}_k\left(x\right)_0 = k$。该算子保存 k 个维度值，

其余置零。误差累积器通过残差项 e_i^t 保留未传输的梯度信息。

4.2.2 梯度压缩算法流程

在明确核心组件后，本文进一步给出动态误差感知的梯度压缩流程。客户端在每轮训练开始时初始化误差计数器与误差累积器，以确保各维度初始选择机会均等。随后，历史残差被叠加至当前梯度形成补偿梯度，提升未传输信息的利用率。根据误差计数器生成自适应概率分布，从而以 Rand-k 策略随机选取部分维度用于上传。动态误差感知的梯度压缩流程如图 4-1 所示。

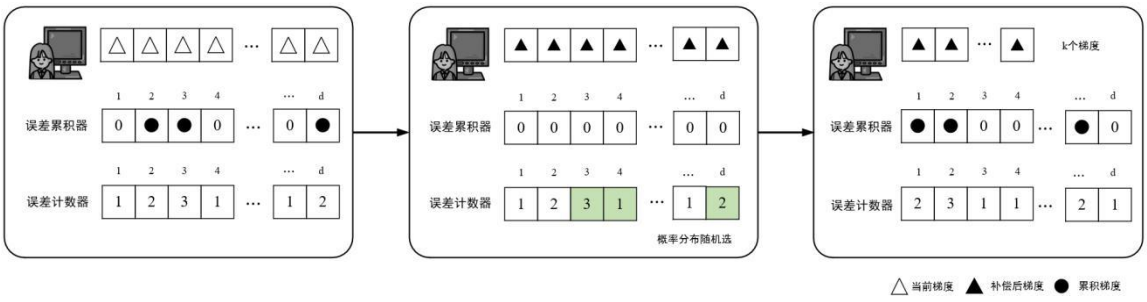


图 4-1 动态误差感知的梯度压缩流程

以下是该算法的详细步骤：

步骤 1：初始化

算法初始化时 $(t=0)$，客户端状态满足：

$$\begin{aligned} c_{i,j}^0 &= 1, \forall i \in \{1,\ldots,d\} \\ e_i^0 &= \mathbf{0} \end{aligned} \tag{4-3}$$

此初始化保证在首次训练时所有维度的选择概率均等，即 $p_{i,j}^1 = \frac{1}{d}$。向量 $\mathbf{0}$ 表示 d 维零向量。

步骤 2：梯度补偿计算

在第 t 轮训练中，客户端计算补偿梯度：

$$\Delta \tilde{w}_i^t = \Delta w_i^t + e_i^{t-1} \tag{4-4}$$

此操作将历史误差 e_i^{t-1} 线性叠加至当前梯度 Δw_i^t，确保前序未传输的梯度分量持续参与参数更新过程。

步骤 3：选择概率生成

基于误差计数器生成维度选择概率分布：

$$p_{i,j}^t = \frac{c_{i,j}^{t-1}}{\sum_{j=1}^{d} c_{i,j}^{t-1}} \tag{4-5}$$

概率分布 $p_i^t \in \mathbb{R}^d$ 满足：

$$\sum_{j=1}^{d} p_{i,j}^t = 1 \tag{4-6}$$

该设计使得长期未被选中的维度 j 因 $c_{i,j}^{t-1}$ 增大而获得更高的选中概率，形成自适应探索机制。

步骤 4：随机维度选择

通过多项式分布采样生成稀疏掩码：

$$\mathcal{C}_i^t \sim \text{Multinomial}\left(k, p_i^t\right) \tag{4-7}$$

其中 $\mathcal{C}_i^t \in \{0,1\}^d$ 满足 $\sum_{j=1}^{d} \mathcal{C}_{i,j}^t = k$。每个掩码元素 $\mathcal{C}_{i,j}^t$ 服从伯努利分布:

$$P\left(\mathcal{C}_{i,j}^t = 1\right) = \frac{k \cdot c_{i,j}^{t-1}}{\sum_{j=1}^{d} c_{i,j}^{t-1}} \tag{4-8}$$

此步骤实现 Rand-k 机制，确保期望选中 k 个维度。

步骤 5：梯度加密操作

对于被选中维度的梯度 $\mathcal{C}_{i,j}^t = 1$，使用 Paillier 公钥进行同态加密：

$$\tilde{c}_{i,j}^t = PaillierEncrypt\left(\mathcal{C}_{i,j}^t \cdot \Delta\tilde{w}_{i,j}^t\right) \tag{4-9}$$

该操作将未选中维度 $\left(\mathcal{C}_{i,j}^t = 0\right)$ 的梯度值强制置零，仅保留 k 个选中维度的原始值，实现梯度加密。

步骤 6：误差残差计算

计算当前轮的未传输梯度残差：

$$e_i^t = \Delta\tilde{w}_i^t - \mathcal{M}_k\left(\Delta\tilde{w}_i^t\right) \tag{4-10}$$

残差向量 e_i^t 包含被压缩舍弃的梯度分量，其将在下一轮训练中通过式（4）重新注入训练过程。

步骤 7：计数器动态更新

误差计数器按以下规则更新：

$$c_{i,j}^t = \mathcal{C}_{i,j}^t \cdot 1 + \left(1 - \mathcal{C}_{i,j}^t\right) \cdot \left(c_{i,j}^{t-1} + 1\right) \tag{4-11}$$

该更新策略实现选中维度 $\left(\mathcal{C}_{i,j}^t = 1\right)$ 重置计数器置 1，未选中维度 $\left(\mathcal{C}_{i,j}^t = 0\right)$ 递增计数器值。此机制动态调整各维度的选择优先级，形成长期公平的探索策略，动态误差感知梯度压缩如算法 4-1 所示。

算法 4-1 动态误差感知梯度压缩算法

输入：当前梯度 Δw_i^t，误差累积器 $\text{accumulator}_i^{t-1}$，误差计数器 counter_i^{t-1}，压缩维度 k

输出：压缩梯度 $\mathcal{M}_k\left(\Delta\tilde{w}_i^t\right)$，更新误差 accumulator_i^t，更新计数器 counter_i^t

if $t=0$ **then**

  for 每个维度 $j\in\{1,\ldots,d\}$ **do**

    $\text{counter}_{i,j}^0 \leftarrow 1$

  end for

  $\text{accumulator}_i^0 \leftarrow \mathbf{0}$

end if

$\Delta\tilde{w}_i^t \leftarrow \Delta w_i^t + \text{accumulator}_i^{t-1}$

计算总和 $S \leftarrow \sum_{k=1}^{d} counter_{i,k}^{t-1}$，采样稀疏掩码 $\sum_{j=1}^{d}\mathcal{C}_{i,j}^t = k$ 且 $P\left(\mathcal{C}_{i,j}^t=1\right)=p_{i,j}^t$

for 每个维度 $j\in\{1,\ldots,d\}$ **do**

$$p_{i,j}^t \leftarrow \frac{\text{counter}_{i,j}^{t-1}}{S}$$

end for

for 每个维度 $j\in\{1,\ldots,d\}$ **do**

  if $\mathcal{C}_{i,j}^t=1$ **then**

    $\tilde{\mathcal{M}}_k\left(\Delta\tilde{w}_i^t\right)_j \leftarrow \Delta\tilde{w}_{i,j}^t$

  else

    $\mathcal{M}_k\left(\Delta\tilde{w}_i^t\right)_j \leftarrow 0$

  end if

end for

$\text{accumulator}_i^t \leftarrow \Delta\tilde{w}_i^t - \mathcal{M}_k\left(\Delta\tilde{w}_i^t\right)$

for 每个维度 $j\in\{1,\ldots,d\}$ **do**

  if $\mathcal{C}_{i,j}^t=1$ **then**

    $\text{counter}_{i,j}^t \leftarrow 1$

  else

    $\text{counter}_{i,j}^t \leftarrow \text{counter}_{i,j}^{t-1}+1$

  end if

end for

返回：压缩梯度 $\mathcal{M}_k\left(\Delta\tilde{w}_i^t\right)$，更新误差 accumulator_i^t，更新计数器 counter_i^t

4.3 安全性分析

（1）抗泄露攻击

泄露攻击是指攻击者针对联邦学习过程中传输的梯度信息进行分析，推断客户端的私有数据，甚而直接复原未经处理的训练数据。普遍存在的泄露攻击涉及梯度逆向攻击以及成员推理攻击，攻击者可凭借多个训练轮次的梯度数据，判定某个特定样本有无参与模型训练，甚至把该样本输入特征进行还原。在加密机制严密性未达标的情形下，攻击者可就梯度的变化走向进行剖析，获得关乎私有数据的敏感信息，由此破坏联邦学习的隐私保护目标达成。

本文通过 Paillier 同态加密和 Rand-k 梯度压缩相结合，有效防御泄露攻击。Paillier 加密保证梯度在加密状态的情景下开展计算，使服务器无法直接探知明文梯度，进而降低梯度逆向攻击的可能性。同时，Rand-k 梯度压缩借助随机选取部分梯度开展传输，减少攻击者可察觉的数据维度。即便攻击者试图恢复数据，也较难获得全面的数据信息。

（2）抗重放攻击

重放攻击是指攻击者在联邦学习过程中截获历史训练轮次的梯度信息并于后续轮次反复提交相同的梯度，以对模型训练的正常开展造成干扰。该种攻击或许会造成模型学习到过时的数据，甚至致使训练过程陷入死循环，对全局模型的正常收敛造成影响。此外，攻击者也许会借助重放攻击对正常的训练行为进行伪造，逃过异常的检测体系，以此实施复杂度更高的攻击。

本文通过 Paillier 同态加密和 Rand-k 梯度压缩相结合，有效防御重放攻击。每次开展 Paillier 加密操作时都会引入随机数，即使把相同的数据加密多次，其加密后的密文也全然不同，从而让攻击者无法简单地重放历史数据。另一方面，Rand-k 梯度压缩机制在每轮训练时随机挑选不同的梯度子集进行上传，致使每一轮训练上传的数据皆不相同。即使攻击者重放历史梯度，亦无法对全局训练过程形成干扰。

（3）抗推理攻击

推理攻击是指攻击者在不直接访问原始数据的情况下，借助模型参数或梯度更新信息，推断训练数据的某些特征或某个数据是否参与了训练。一般的推理攻击包含成员推理攻击，即攻击者尝试推测某个样本是否在训练集中；以及属性推理攻击，即攻击者企图复原某个训练样本的特征数值。在传统联邦学习框架下，服务器具备观察客户端上传梯度更新的能力，所以存在较大的信息泄露隐患。

本文通过 Paillier 同态加密和 Rand-k 梯度压缩相结合，有效防御推理攻击。Paillier 加密使得服务器无法直接观察梯度的真实数值，即便攻击者截获梯度，也

不能直接开展推理分析。同时，Rand-k 梯度压缩在每轮训练中随机选择不同的梯度子集开展传输，使攻击者无法构建一套完整的梯度变化模式，从而进一步降低推理攻击成功的概率。

（4）抗投毒攻击

投毒攻击是指恶意参与者在联邦学习过程中，蓄意操纵或污染其提交的训练数据与梯度，造成全局模型性能下降，甚至在特定情况里引入后门，引发预测结果出现差错。投毒攻击主要分为数据投毒和梯度投毒两种类型。数据投毒是指攻击者在其本地数据集注入恶意样本，例如修改标签的数据信息或添加特殊触发模式，导致全局模型在攻击者选定的场景下呈现异常表现；梯度投毒则是指攻击者在本地训练后，对上传的梯度加以篡改，例如放大某些特征权重，让模型在处理特定样本时产生偏差，引起目标类别的错误预测。

本文通过 Paillier 同态加密和 Rand-k 梯度压缩相结合，有效防御投毒攻击。Rand-k 梯度压缩随机挑选部分梯度上传，让恶意节点难以精确统筹全局梯度变化格局，减弱投毒攻击的有效性。此外，误差补偿机制保证未上传的梯度信息不会被彻底丢弃，而是累积至后续轮次，进而降低梯度压缩造成的信息损失，提升模型的鲁棒性。

4.4 收敛性分析

在开始收敛性分析前，需要以下定理保障算法在迭代步骤里能以特定速率逼近最优解，并对误差的稳态界进行量化：

定理 1（L－平滑性[79]） 全局目标函数 $L(w)$ 满足 L －Lipschitz 连续梯度条件：

$$\left\|\nabla L(w)-\nabla L(w')\right\| \le L\left\|w-w'\right\| \tag{4-12}$$

其中 w' 为任意模型参数向量。此条件确保梯度的变化速率受常数 L 限制。

定理 2（ μ －强凸性[80]） 目标函数 $L(w)$ 满足 μ －强凸性：

$$L(w) \ge L(w')+\nabla L(w')^{\top}(w-w')+\frac{\mu}{2}\left\|w-w'\right\|^2 \tag{4-13}$$

其中 w' 为任意参数点。此性质保证存在唯一全局最优解 w^* 。

定理 3（梯度有界性[81]） 客户端本地梯度 Δw_i^t 的幅度满足：

$$\left\|\Delta w_i^t\right\|_2 \le G \tag{4-14}$$

其中 G 为梯度上界常数。此约束防止本地更新量过大导致系统不稳定。

定理 4（压缩算子性质[82]） 压缩算子 $\mathcal{M}_k$ 满足：

$$\mathbb{E}\left[\left\|\mathcal{M}_k(x)-x\right\|^2\right]\le\left(1-\frac{k}{d}\right)\|x\|^2 \tag{4-15}$$

其中 $x\in\mathbb{R}^d$ 为任意输入向量。此不等式量化了梯度压缩的期望信息损失率。

若要证明方案收敛，需证明以下不等式成立：

$$\mathbb{E}\left[\left\|w^t-w^*\right\|^2\right]\le\left(1-\frac{\mu^2}{4L^2}\right)^t\left\|w^0-w^*\right\|^2+\frac{2G^2d}{\mu^2k} \tag{4-16}$$

若该式成立，则算法在任意初始点 w^0 下能以线性速率收敛至稳态误差界 $\frac{G^2d}{\mu k}$，证明方案有效。

以下是收敛性证明：

定义 Lyapunov 函数以统一分析参数误差和残差累积：

$$V^t=\left\|w^t-w^*\right\|^2+\frac{\eta}{\mu}\sum_{i=1}^{n}\left\|e_i^t\right\|^2 \tag{4-17}$$

其中 $\left\|w^t-w^*\right\|^2$ 是参数误差项，直接衡量优化进度。$\frac{\eta}{\mu}\sum_{i=1}^{n}\left\|e_i^t\right\|^2$ 是残差惩罚项，量化未传输梯度的累积效应。

参数更新规则为：

$$w^{t+1}=w^t-\eta\nabla L\left(w^t\right) \tag{4-18}$$

展开参数误差平方范数可得：

$$\begin{aligned}\left\|w^{t+1}-w^*\right\|^2&=\left\|w^t-\eta\nabla L\left(w^t\right)-w^*\right\|^2\\&=\left\|w^t-w^*\right\|^2-2\eta\left\langle\nabla L\left(w^t\right),w^t-w^*\right\rangle+\eta^2\left\|\nabla L\left(w^t\right)\right\|^2\end{aligned} \tag{4-19}$$

又由强凸性（定理 2），取 $w=w^*$，$w'=w^t$，代入得：

$$\left\|w^{t+1}-w^*\right\|^2\le(1-\mu\eta)\left\|w^t-w^*\right\|^2+\eta^2\left\|\nabla L\left(w^t\right)\right\|^2 \tag{4-20}$$

应用平滑性（定理 1），由于 $L(w^*)$ 为 0，化简得：

$$\left\|\nabla L\left(w^t\right)\right\|\le L\left\|w^t-w^*\right\| \tag{4-21}$$

对上式进行两边平方后代入式（4-19）：

$$\left\|w^{t+1}-w^*\right\|^2\le\left(1-\mu\eta+\eta^2L^2\right)\left\|w^t-w^*\right\|^2 \tag{4-22}$$

此时 $\eta=\frac{\mu}{2L^2}$ 为最小值点，则：

$$\begin{aligned}1-\mu\eta+\eta^2L^2 &=1-\mu\cdot\frac{\mu}{2L^2}+\left(\frac{\mu}{2L^2}\right)^2L^2\\ &=1-\frac{\mu^2}{2L^2}+\frac{\mu^2}{4L^2}\\ &=1-\frac{\mu^2}{4L^2}\end{aligned} \tag{4-23}$$

因此：

$$\left\|w^{t+1}-w^*\right\|^2\le\left(1-\frac{\mu^2}{4L^2}\right)\left\|w^t-w^*\right\|^2 \tag{4-24}$$

由残差定义和定理 4，对 e_i^t 取期望可得：

$$\mathbb{E}\left[\left\|e_i^t\right\|^2\right]\le\left(1-\frac{k}{d}\right)\mathbb{E}\left[\left\|\Delta\tilde{w}_i^t\right\|^2\right] \tag{4-25}$$

平方展开可得：

$$\begin{aligned}\mathbb{E}\left[\left\|\Delta\tilde{w}_i^t\right\|^2\right]&=\mathbb{E}\left[\left\|\Delta w_i^t+e_i^{t-1}\right\|^2\right]\\ &=\mathbb{E}\left[\left\|\Delta w_i^t\right\|^2\right]+\mathbb{E}\left[\left\|e_i^{t-1}\right\|^2\right]+2\mathbb{E}\left[\left\langle\Delta w_i^t,e_i^{t-1}\right\rangle\right]\end{aligned} \tag{4-26}$$

由于 Δw_i^t 是第 t 轮计算的本地梯度，与历史残差 e_i^{t-1} 独立，且 $\mathbb{E}\left[\Delta w_i^t\right]=0$，由此可得：

$$\mathbb{E}\left[\left\langle\Delta w_i^t,e_i^{t-1}\right\rangle\right]=\mathbb{E}\left[\mathbb{E}\left[\left\langle\Delta w_i^t,e_i^{t-1}\right\rangle|\ e_i^{t-1}\right]\right]=0 \tag{4-27}$$

由定理 3，可得：

$$E\left\|\Delta w_i^t\right\|^2\le G^2 \tag{4-28}$$

因此可得递推式：

$$\mathbb{E}\left[\left\|e_i^t\right\|^2\right]\le\left(1-\frac{k}{d}\right)\left(\mathbb{E}\left[\left\|e_i^{t-1}\right\|^2\right]+G^2\right) \tag{4-29}$$

逐层展开式（4-28），可得：

$$\begin{aligned}\mathbb{E}\left[\left\|e_i^t\right\|^2\right] &\le \left(1-\frac{k}{d}\right)\left(E\left[\left\|e_i^{t-1}\right\|^2\right]+G^2\right)\\ &\le \left(1-\frac{k}{d}\right)^2 E\left[\left\|e_i^{t-2}\right\|^2\right]+\left(1-\frac{k}{d}\right)G^2+\left(1-\frac{k}{d}\right)G^2\\ &\vdots\\ &\le \left(1-\frac{k}{d}\right)^t E\left[\left\|e_i^0\right\|^2\right]+G^2\left(1-\frac{k}{d}\right)\sum_{l=0}^{t-1}\left(1-\frac{k}{d}\right)^l\end{aligned} \tag{4-30}$$

因 $\mathbb{E}\left[\left\|e_i^0\right\|^2\right]=0$，化简为：

$$\mathbb{E}\left[\left\|e_i^t\right\|^2\right] \le G^2\left(1-\frac{k}{d}\right)\sum_{l=0}^{t-1}\left(1-\frac{k}{d}\right)^l \tag{4-31}$$

令 $r=1-\frac{k}{d}$，可得：

$$\sum_{l=0}^{t-1}\left(1-\frac{k}{d}\right)^l=\frac{1-\left(1-\frac{k}{d}\right)^l}{\frac{k}{d}} \tag{4-32}$$

因此有：

$$\begin{aligned}\mathbb{E}\left[\left\|e_i^t\right\|^2\right] &\le G^2\left(1-\frac{k}{d}\right)\cdot\frac{1-\left(1-\frac{k}{d}\right)^t}{\frac{k}{d}}\\ &=\frac{G^2 d}{k}\left(1-\left(1-\frac{k}{d}\right)^{t+1}\right)\end{aligned} \tag{4-33}$$

当 $t\to\infty$ 时，$\left(1-\frac{k}{d}\right)^t\to 0$，故：

$$\lim_{t\to\infty}\mathbb{E}\left[\left\|e_i^t\right\|^2\right]\le\frac{G^2 d}{k} \tag{4-34}$$

将式（4-23）和式（4-33）代入 $V^{(t)}$：

$$\begin{aligned}\mathbb{E}\left[V^t\right]&=\mathbb{E}\left[\left\|w^t-w^*\right\|^2\right]+\frac{\eta}{\mu}\sum_{i=1}^{n}\mathbb{E}\left[\left\|e_i^t\right\|^2\right]\\&\le\left(1-\frac{\mu^2}{4L^2}\right)\mathbb{E}\left[\left\|w^{t-1}-w^*\right\|^2\right]+\frac{\eta}{\mu}\cdot\frac{nG^2d}{k}\\&\le\left(1-\frac{\mu^2}{4L^2}\right)^t\left\|w^0-w^*\right\|^2+\frac{\eta nG^2d}{\mu k}\sum_{l=0}^{t-1}\left(1-\frac{\mu^2}{4L^2}\right)^l\end{aligned}\tag{4-35}$$

将几何级数和代入联合能量函数：

$$\mathbb{E}\left[V^t\right]\le\left(1-\frac{\mu^2}{4L^2}\right)^t\left\|w^0-w^*\right\|^2+\frac{\eta nG^2d}{\mu k}\cdot\frac{4L^2}{\mu^2}\left(1-\left(1-\frac{\mu^2}{4L^2}\right)^t\right)\tag{4-36}$$

当$t\to\infty$时，$\left(1-\frac{\mu^2}{4L^2}\right)^t\to 0$，稳态项为：

$$\frac{\eta nG^2d}{\mu k}\cdot\frac{4L^2}{\mu^2}=\frac{4\eta nG^2dL^2}{\mu^3k}\tag{4-37}$$

将$\eta=\frac{\mu}{2L^2}$代入稳态项：

$$\begin{aligned}\frac{4\eta nG^2dL^2}{\mu^3k}&=\frac{4\cdot\frac{\mu}{2L^2}\cdot nG^2dL^2}{\mu^3k}\\&=\frac{2nG^2d}{\mu^2k}\end{aligned}\tag{4-38}$$

并归一化$n=1$，最终形式为：

$$\mathbb{E}\left[V^t\right]\le\left(1-\frac{\mu^2}{4L^2}\right)^t\left\|w^0-w^*\right\|^2+\frac{2G^2d}{\mu^2k}\tag{4-39}$$

由于$V^t\ge w^t-w^{*2}$（残差项非负），最终参数误差满足：

$$\mathbb{E}\left[\left\|w^t-w^*\right\|^2\right]\le\left(1-\frac{\mu^2}{4L^2}\right)^t\left\|w^0-w^*\right\|^2+\frac{2G^2d}{\mu^2k}\tag{4-40}$$

由此得证。

4.5 通信开销分析

在联邦学习训练过程中，通信成本通常是影响系统效率的主要瓶颈之一。每轮迭代中，客户端需将本地模型的梯度上传至聚合节点，若模型参数维度为d，则

一次完整上传需传输 d 个浮点数。在进行 T 轮训练的情况下，若无任何压缩措施，总通信量为 Td 。

在采用稀疏梯度压缩策略后，客户端仅需上传 k 个非零梯度值（其中 k 远小于 d ），从而使每轮上传通信量降低为 k 。设压缩率为 $r=\frac{k}{d}$，则每轮通信量为 rd ，累计通信量为 Trd ，节省的通信总量为 $(1-r)Td$ ，具有显著的带宽减负效应。

此外，若以实际应用中常见的轮数为 10（T=10）为参考，结合不同压缩率的设定，可对通信量的节省程度进行估算，T=10 下的不同压缩率设定下的通信开销如表 4-1 所示。

表 4-1 不同压缩率设定下的通信开销（T=10）

压缩率	每轮上传维度数	未压缩通信量	当前通信量	累计节省通信量
不压缩	d	10d	10d	0
0.5	0.5d	10d	5d	5d
0.1	0.1d	10d	1d	9d
0.01	0.01d	10d	0.1d	9.9d

由表中数据可见，即使是在极低压缩率（如 1%）下，系统仍能保留完整训练流程的核心信息，同时将通信量压缩至原始的 1%，对于部署在边缘设备或低带宽环境中的联邦学习任务具有重要价值。

需要说明的是，由于引入了误差反馈机制，未被选择上传的梯度在本地仍会被保存，并在后续训练轮次中得到补偿，因此不会因压缩而长期丢失有价值的梯度信息，通信效率与模型收敛之间可实现良好平衡。

4.6 实验评估

4.6.1 实验准备

本章实验的硬件环境与软件配置和第三章保持一致，无需重复描述。实验主要采用 MNIST 和 Fashion-MNIST 数据集，并采用 CNN 网络进行模型训练。

本实验使用的 CNN 网络包含两层卷积层和一层全连接层，其中卷积层的目的是提取图像特征，全连接层执行分类任务。相比于深度网络，该结构计算量不高，更适用于评估梯度压缩对通信效率和模型性能的影响。

4.6.2 实验设计

本章围绕提出的梯度压缩算法及同态加密方案展开实验分析，核心目标是检

验梯度压缩对联邦学习的通信效率及模型性能的影响。已有研究证实，联邦学习上传的梯度信息中存在大量冗余部分，其中一些梯度对模型优化的贡献不大，造成通信开销增加。因此，适宜的梯度压缩策略不但能降低带宽消耗，还可维持模型性能的稳定性，从而提升联邦学习的整体效率。

为验证本文采用方法的有效性，实验重点关注梯度压缩在不同设定模式下的影响，且对其影响模型收敛性和准确率的情况进行测评。基于此，本文设计了如下实验：

实验一：针对本文提出的梯度压缩算法，分析在 MNIST、Fashion-MNIST 数据集下，不同 batch size 设定对模型性能的影响。

实验二：针对本文提出的梯度压缩算法，分析在 MNIST、Fashion-MNIST 数据集下，不同压缩率设定对模型性能的影响。

4.6.3 实验参数设置

本章实验的联邦学习设置与第三章保持一致，包括客户端数量（total_clients=30）、每轮选取客户端数（per_round_workers=15）、全局训练轮次（global_epochs=100）和本地训练轮次（local_epochs=3）。此外，同态加密的密钥长度设定为 1024 位（key_bits=1024），以确保梯度加密的安全性。

实验一：批次大小（batch_size）分别设定为 16、32、48、64，用于评估不同批次大小对梯度压缩策略的影响。

实验二：压缩率（compression_ratio）分别设定为 1%、0.5%、0.1%、0.01%，用于分析不同梯度压缩率对通信开销和模型性能的影响。

4.6.4 实验结果分析

实验一：在 MNIST 数据集上，不同 batch size 设定对模型的收敛速度和最终准确率均有一定影响。其中，batch size 为 16 和 32 的情况下，收敛较慢，在 35 轮左右达到稳定。而 batch size 为 48 和 64 时，收敛速度更快，在 30 轮左右即可完成收敛。MNIST 数据集下不同 Batch Size 设定的性能比较如图 4-2 所示。

尽管收敛速度存在差异，但四种 batch size 设定最终的准确率较为接近，四种 batch size 由小到大的准确率分别为 97.62%、97.12%、96.27%、97.1%。考虑到收敛速度与准确率的平衡，batchsize 设定为 64 能够在保持较高准确率的同时提高收敛速度，因此在本算法中更具优势。

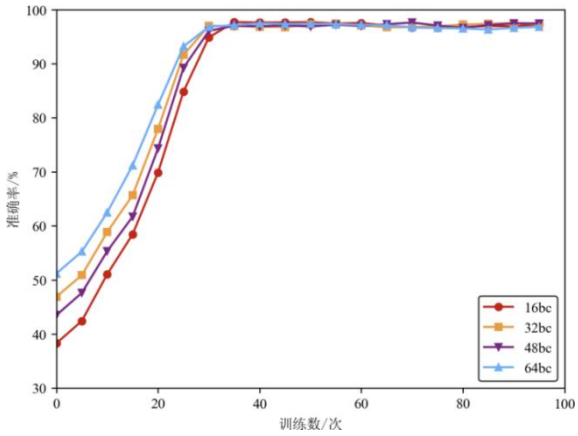


图 4-2 MNIST 数据集下不同 Batch Size 设定的性能比较

在 Fashion-MNIST 数据集上，batch size 的选择对收敛轮数影响较大。其中，batch size 为 16 的模型因批次较小，收敛速度最慢，在 78 轮左右稳定，最终准确率为 92.45%。batch size 为 32 的模型在 70 轮完成收敛，准确率为 92.47%。batch size 为 48 的模型在 57 轮收敛，准确率为 92.18%，而 batch size 为 64 的模型在 42 轮就达到收敛，最终准确率为 92.4%。Fashion-MNIST 数据集下不同 Batch Size 设定的性能比较如图 4-3 所示。

综合来看，batch size 为 64 的设定不仅收敛最快，同时保证了较高的模型准确率，因此在本算法中表现最佳。

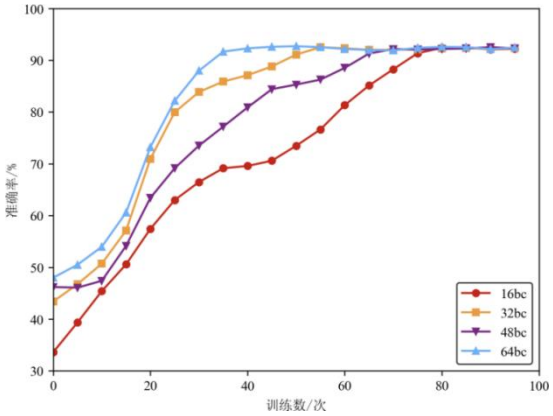


图 4-3 Fashion-MNIST 数据集下不同 Batch Size 设定的性能比较

实验二：在 MNIST 数据集上，不同压缩率（compression_ratio）对收敛速度和最终准确率影响较大。压缩率为 0.01%时的收敛最快，在 27 轮左右完成训练，但其最终准确率较低，仅 88.51%，表明过度压缩损害了模型性能。0.5%和 1%的压缩率收敛速度接近，在 30 轮左右达到稳定，其中 0.5%的压缩率表现最佳，最终准确率达到 97.1%，而 1%的压缩率准确率为 93.1%，表现略差。0.1%的压缩率收

敛最慢，在 46 轮才完成训练，最终准确率为 93.4%。MNIST 数据集下不同压缩率设定的性能比较如图 4-4 所示。

综合来看，0.5%压缩率在保证较高准确率的同时保持了较快的收敛速度，因此是 MNIST 数据集上的最优选择。

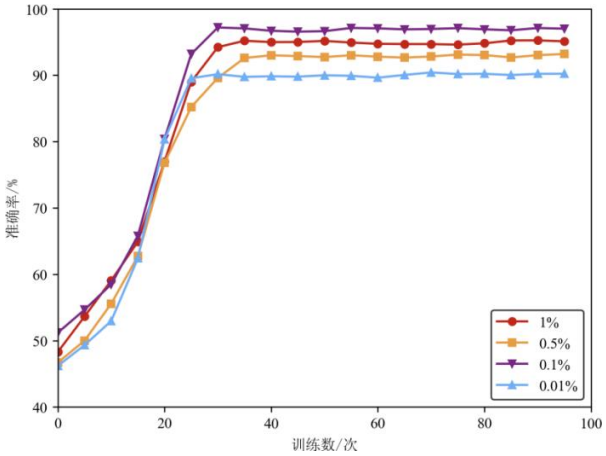


图 4-4 MNIST 数据集下不同压缩率设定的性能比较

在 Fashion-MNIST 数据集上，压缩率为 0.01%时达成最快收敛，在 25 轮左右完成训练，但最终准确率仅 76.9%，表明过度压缩导致模型性能下降。0.1%压缩率的准确率较高为 88%，但收敛较慢，在 57 轮左右才稳定。0.5%压缩率收敛最快，在 39 轮完成训练，并达到最高准确率 92.4%。1%压缩率略逊于 0.5%，在 43 轮完成收敛，最终准确率为 90.3%。Fashion-MNIST 数据集下不同压缩率设定的性能比较如图 4-5 所示。

综合来看，0.5%压缩率在 Fashion-MNIST 数据集上同样是最优选择，兼顾收敛速度和最终准确率。

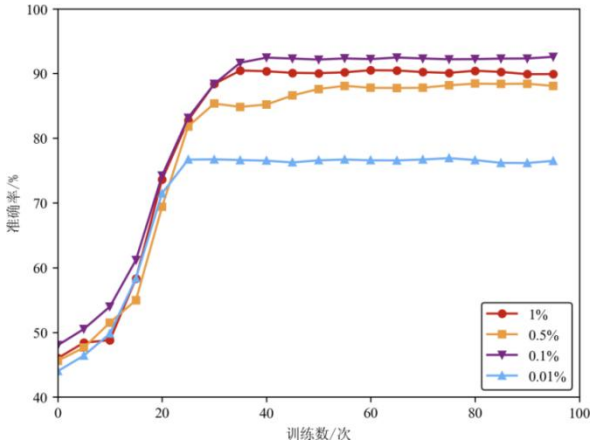


图 4-5 Fashion-MNIST 数据集下不同压缩率设定的性能比较

4.7 本章小结

本章提出了一种结合动态误差感知与同态加密的梯度压缩算法，兼顾通信效率与模型隐私保护。在算法结构与安全性分析的基础上，进一步通过实验验证了所提方法在不同参数设定下的训练性能，展现出良好的压缩效果与模型收敛表现。

第五章 基于区块链的隐私增强联邦学习系统设计与实现

5.1 引言

随着联邦学习的应用逐步扩大，对训练过程的可管理性、贡献评估和操作便捷性的需求不断增强。为此，本章设计并构建了一套基于区块链的联邦学习系统，该系统包含模型训练、训练管理、贡献管理、用户管理以及日志记录等功能，其目标是打造一个高效、透明化的联邦学习训练与管理环境。

该系统运用可视化界面，用户可以快捷完成模型训练参数配置、加密方式选择、贡献值查询等操作，同时对训练进度及结果开展实时监控。此外，训练完成后可呈现客户端在联邦学习中的总贡献值、当选领导人节点次数，亦支持贡献值排序、查询任务，为系统公平性评估提供直观依据。系统也支持查看、管理及下载训练信息，并且提供日志功能，帮助用户跟踪训练走向。

本章将针对系统的整体架构、功能设计与实现逻辑进行分析，核实其在联邦学习任务管理与贡献评估方面的可行性，为后续的研究和应用提供支撑。

5.2 系统概述

基于区块链的联邦学习系统是一套支持模型训练、训练管理、贡献评估以及用户操作的联邦学习平台，其目的是实现高效、透明和可追溯的训练管理环境。该系统的主要功能是针对联邦学习训练任务开展配置及执行，便于用户跟踪训练实施进程，并针对各客户端的贡献情形开展分析。

系统采用模块化的设计方式，涉及模型训练、训练管理、贡献管理、用户管理和日志信息五大核心功能模块，用户在系统里可配置训练参数，并借助可视化界面实时监控训练进度与结果。训练结束后，系统实行对训练记录的管理，支持对训练结果做查询、修改、下载处理，并与贡献管理模块相结合，直观呈现客户端的本轮贡献值、总贡献值、当选领导人节点次数等关键信息。用户可以凭借训练名称、训练编号、训练轮次等信息实施贡献值的查询，亦支持贡献值排序、重置查询和贡献数据下载任务，进而对各客户端的长期贡献及训练表现加以评估。

本系统适用于联邦学习实验和隐私增强模型的管理工作，为研究人员和开发者提供了可视化、简单易用且具备贡献评估能力的训练管理工具，利于开展联邦学习实验、模型优化及公平性分析。

5.3 系统需求分析

为保证本系统可满足用户的实际需求，并拥有良好的可用性和稳定性，本文从用户需求分析、功能需求分析和性能需求分析三个方面对系统的需求做详细说明。

5.3.1 用户需求分析

该系统是一个基于区块链的隐私增强联邦学习系统，主要用于多方数据协作、联邦学习模型训练及贡献评估。依据用户的身份和职责差异，系统的用户主要分为管理员和普通用户。不同类型的用户在操作系统时具备不同的功能诉求。

管理员作为系统的管理者，担负系统维护、用户管理、训练任务管理及贡献评估等事务，保障系统稳定执行、安全无患且数据管理规范严谨。在系统运行层面，管理员需要对系统参数实施管理，保证联邦学习任务顺利开展，并对系统状态开展实时监测，以防止异常行为对整体系统稳定性造成干扰；围绕用户权限控制层面，管理员应被赋予用户管理权限，即可以添加、修改和删除系统用户，保证普通用户只能对自己的数据实施访问与管理，而管理员可针对全体用户实施管理；与此同时，管理员在训练数据监管方面也具备重要权限。为实现系统正常运行的既定目标，管理员需要查询与监管所有用户的训练记录，且具备删除训练记录的权限，以维护数据的实际有效；管理员也需处置贡献管理方面的事务，能够查看所有客户端的贡献相关数据，包括总贡献值、当选领导人节点次数等信息，且能依据训练名称、训练编号或轮次查询贡献情况，以便进一步有针对性地开展对贡献评估结果的分析；此外，管理员需要访问训练日志，监控系统的运行状态，且在必要的时候删除无效或过期的日志，保障系统稳定运行。

（2）普通用户的需求

普通用户是系统的主要使用人群，重点围绕模型训练、训练结果管理、贡献评估及日志信息查询展开。用户应能选择合适的模型和数据集，自行设置训练所需的参数，如训练轮数、批量大小、学习率等，并开启训练任务；在训练起始阶段，用户应能实时监控训练进展，待训练完成后查看系统生成的训练记录，包括训练名称、准确率、使用的数据集以及模型等关键信息，同时用户应能对训练名称及备注做修改，进而便于自身针对训练数据进行管理，下载训练成果，以便进一步针对模型表现开展多方面分析；在贡献评估方面，用户应能查看自身在每轮训练中的贡献值、历史贡献值以及是否当选领导人节点，并通过贡献值调整本地训练策略；此外，普通用户还应具备训练日志访问能力，可用于追踪训练过程中的系统响应与任务状态，及时发现潜在异常，提升任务可控性与透明度。

5.3.2 功能需求分析

本系统是一个基于区块链的隐私增强联邦学习系统，主要包含模型训练、训练管理、贡献管理、用户管理和日志信息这五大核心功能。这些功能模块共同构成了完整的联邦学习平台，保障用户能够高效地开展模型训练与管理，进而增强数据的安全性与系统的稳定性。

模型训练是系统的核心功能，包括参数配置、训练启动、训练状态监控等任务。在开展模型训练前，系统需赋予用户在训练前拥有设定模型参数、学习参数和加密参数的权限。在模型开展训练的过程中，系统应实时反馈训练的进展情形，呈现当前轮次的准确率和损失值情况，帮助用户直观地了解模型的训练成效。用户还可以在训练过程中选择中止训练，系统应自动把现有的训练记录留存，并存储于日志信息页面，以便后续进行分析及优化。

训练管理则把重点放在对用户的历史训练信息实施存储和管理，拥有便利的数据查询与记录维护功能。系统应支持用户通过数据集、模型类型、发起人等条件筛选训练记录的能力，并且能够查看训练的详细信息。用户可针对自己的训练记录实施修改，例如修改训练名称或备注，以便开展后续的管理和分析。管理员除享有全部查询、修改权限以外，也能够删除训练记录，以保障系统数据的规范性与有效性。

贡献管理负责展示各客户端的贡献情况，包括总贡献值、当选领导人节点次数等关键信息，同时在训练结束后呈现本轮贡献值排行以及历史贡献值排行，保证贡献评估的透明性。用户可按照训练名称、训练编号及训练轮次查询贡献情况，且支持依据贡献值的大小进行排序，直观分析不同客户端在训练期间的贡献。系统应当支持贡献数据下载功能，便于开展进一步分析和存档。系统还添加查询重置功能，实现数据管理的便捷性。管理员可对所有用户的贡献情况进行整体把控，维护系统的公平性与贡献评估的可追溯性。

用户管理主要进行系统用户的管理以及权限控制功能，以保障系统兼顾可操作性与安全性。普通用户有权对所有账户信息进行查看，且能够对自己的密码、邮箱和电话号码等进行修改，而管理员拥有更高级的权限，包括添加、修改和删除用户信息，让系统的用户数据更新至最新状态。管理员可对用户的角色权限实施管理，让普通用户仅可管理自己的数据，而自身具备全局管理的能力，保障系统的公平性与可控性。

日志信息主要针对系统内的关键操作信息予以记录。用户可搜索全部的训练日志，并查看日志的训练详情，包含训练过程中的配置参数、训练状态以及全面的训练流程描述。为了让用户快速查找日志，系统给出多样化的筛选条件，如数

据集、模型类型、发起人等。管理员则拥有删除日志的权限，可以清理过期或无效的日志数据，以实现系统存储管理的优化。

考虑到管理员在系统中承担全局性管理职责，其权限范围覆盖模型训练、训练管理、贡献管理、用户管理与日志信息等全部功能模块。即便某些功能如模型训练并非常规操作项，管理员仍需具备相应权限，以便在系统测试或权限核查等场景中灵活调用相关功能。相比于普通用户的功能权限通常局限于自身数据与操作范围，管理员更侧重于对全局状态的掌控与跨模块的信息管理。

基于上述权限覆盖关系，系统在用例图设计中采用泛化关系对用户角色进行建模，将管理员作为普通用户的扩展角色进行定义。该设计不仅保留了角色之间的继承逻辑，也使系统权限层次更为清晰。在此结构下，管理员继承了普通用户的基本功能操作路径，同时引入了若干面向系统级管理的高级操作，如用户账户增删、日志信息清理、训练数据维护及贡献评估监管等。该设计体现了系统权限结构的层次性与逻辑一致性，有助于清晰划分用户职责并规范权限控制机制。总体用例图如图 5-1 所示：

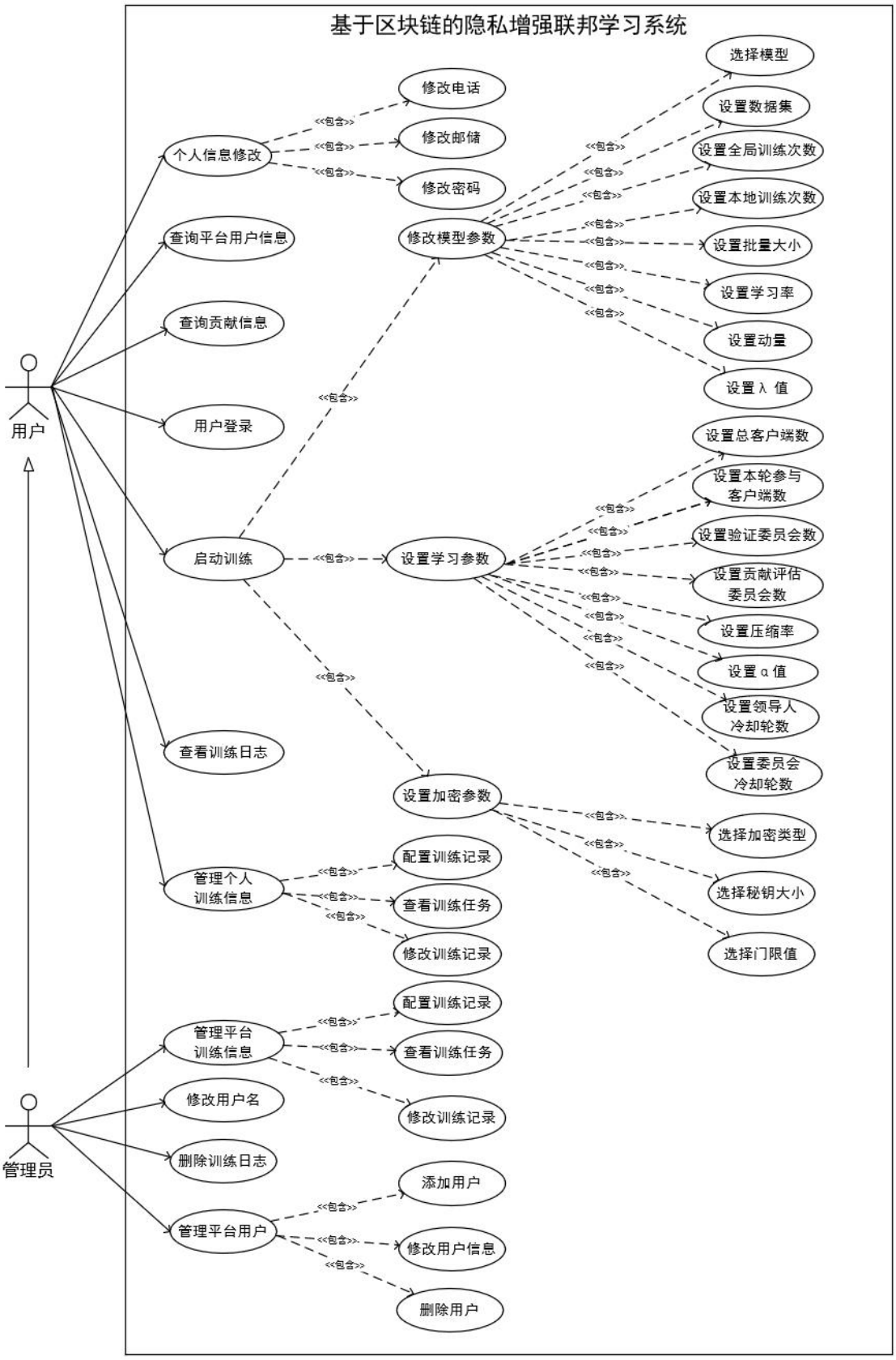


图 5-1 总体用例图

5.3.3 性能需求分析

在系统达成基本功能需求后，为使用户可以顺利使用系统，且增进整体使用便捷度，需保障系统的相关性能。尽管本系统的功能较为基础，但仍需在以下几个方面达成基本的性能需求，以保障系统能够稳定运行并给予用户良好体验。

（1）系统稳定性

该系统支持多个用户同时使用，涉及模型训练、训练管理、贡献管理、用户管理及日志信息功能。为了让系统在正常情况下可稳定运行，应保障用户在实施训练任务、查询数据以及管理用户等阶段，不会因系统故障导致操作出现失败的状况。系统采用常见的数据库存储方式，确保数据不会因异常中断而丢失。同时在训练过程实施阶段给予状态的反馈，保证用户随时可对训练进程进行查看。如果训练过程中出现异常情况，系统可把训练中出现的错误信息记录，有助于后续故障排查工作。

（2）数据存储可靠性

系统的核心数据由训练记录、贡献信息、用户信息和日志信息构成，这些数据的存储需要保障完整性和可查询性。用户需要有随时查询自己训练记录的权限，并取得对应的训练结果，而管理员需要对用户及训练数据加以管理，因此数据库的存储和访问需要维持稳定。系统采用关系型数据库进行存储，并提供简单的数据备份机制，防止因系统异常造成数据丢失。此外日志管理功能可以把系统运行状态记录下来，方便管理员对系统运行状况加以监控。

（3）系统响应速度

本系统的功能主要涉及用户对数据库的相关操作，因此系统的响应速度需维持在合理范围内。系统采用常见的数据查询优化方式，减少数据库查询阶段的时间消耗，保证用户在操作系统期间能够快速获取所需信息。为实现页面加载效率的提升，系统采用基础的前端缓存机制，减少重复的请求，增进用户体验感。从训练任务的进度展示来看，系统赋予简单的实时反馈机制，用户可直观地查看训练进度，而不必等到训练结束后再去查看结果。

（4）系统并发性

本系统主要针对单个组织以及小规模用户的使用偏好，通常不会有大量用户同时进行操作，因此并发性要求较低。但为了保证多个用户可在同一时间访问系统，系统仍然支持基本的多用户并发访问。例如多个用户可同时开展针对训练记录的查询，或者管理员在用户使用系统的阶段同步管理用户信息。就数据库访问而言，系统运用基础的事务管理，保证多个用户同时操作时数据不会面临冲突或丢失的现象。

（5）系统易用性

本系统针对的用户群体可能不具备深厚的技术背景，因此系统的界面设计应当简单直观。在训练任务的配置阶段中，系统给出清晰的输入框和选择框，保证用户可以直观地完成参数设定，减少用户的实施操作的难度。同时用户管理和训练管理页面的查询功能支持模糊查询，用户可便捷高效地筛选与查找所需信息。此外日志管理页面呈现了详细的训练过程信息，便于用户查看系统的运行状况，而不用手动记录。

5.4 系统设计

5.4.1 系统架构设计

本系统采用分层架构设计，把系统功能拆分为模型层、视图层、控制层和算法层，各层，各层相互协作，以保证系统功能清晰、可扩展性强，且能满足不同用户的需求，系统架构设计图如图 5-2 所示。

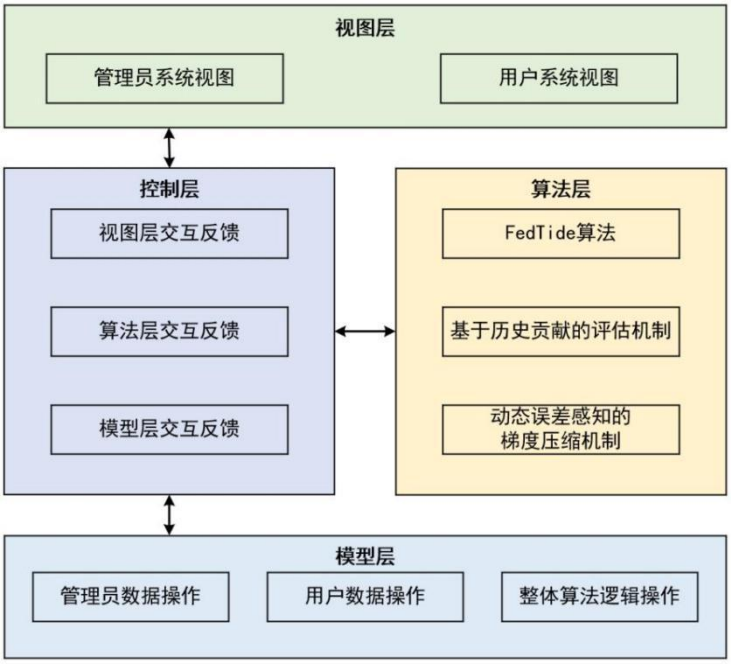


图 5-2 系统架构设计图

模型层作为系统的数据管理核心，承担着系统里用户信息、训练信息、贡献信息、日志信息以及模型相关参数的存储与维护工作，该层赋予数据增删改查的功能，并保证数据在系统里的一致性和完整性。模型层向其他层提供数据支持，是系统稳定运行的关键支撑。

视图层是系统的用户交互界面，承担接收用户输入并呈现系统功能的职责。

该层包含模型训练、训练管理、贡献管理、用户管理和日志信息等主要界面，让用户能够直观地对该系统进行操作。视图层借助调用控制层相关功能，实现用户与系统的交互操作，还把系统反馈的信息以直观的方式呈递给用户。

控制层承担系统的核心业务逻辑，负责应对视图层提出的请求，并开展模型层和算法层之间的交互。该层实施对用户权限、训练信息、贡献信息及日志信息的管理，让不同用户的操作权限与系统规则达成一致，并维持系统的正常工作。控制层还提供一系列接口，保证系统的各个功能模块可以顺畅衔接，并维护系统整体的长期稳定性。

算法层作为系统的计算核心，承担着开展模型训练、参数优化、梯度处理、贡献评估及数据加密等任务。用户于视图层配置训练参数后，控制层将任务呈交给算法层实行核算。当算法层完成训练任务后，把结果回传给控制层，且由视图层呈现给用户。该层确保系统具备联邦学习的基本能力，且支持不同的训练策略，以实现不同应用的要求。

本系统所采用的分层架构让各层功能独立，实现了系统的模块化设计，增强了系统的可扩展性与维护性。各层通过接口开展交互，让系统拥有能力灵活应对多样应用场景的能力，并为未来功能的扩充起到支撑作用。

5.4.2 系统功能设计

为实现系统在多角色共同参与下高效、有序地运行，本系统在功能设计阶段遵循“权限分层、职责划分”的原则，依照不同角色的使用需求展开模块化功能规划，就整体来说系统功能可划分成两个主要部分：管理员端功能设计与用户端功能设计。

管理员端面向系统的全局运维与权限控制场景，覆盖模型训练、训练管理、贡献管理、用户管理与日志信息五大功能模块。管理员除具备对训练任务参数进行配置及控制的能力外，还可对用户数据、训练记录及贡献信息进行综合性管理，也可借助日志管理实现系统状态的持续监督与维护。该端功能设计把系统的安全性、稳定性与数据一致性作为重点考量，维持联邦学习平台在多用户协同工作环境下的有序性。管理员端功能设计如图 5-3 所示。

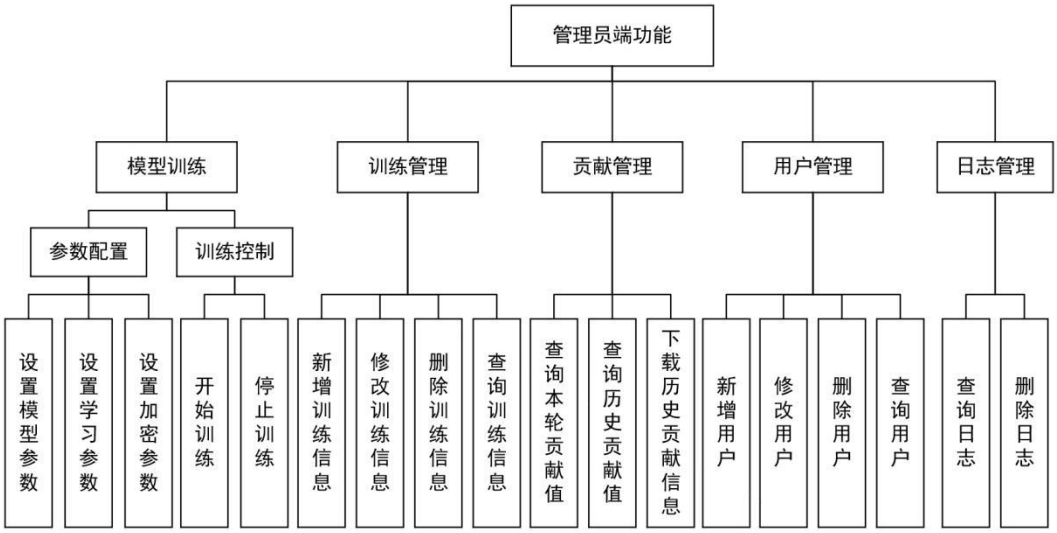


图 5-3 管理员端功能设计

用户端聚焦于模型训练与个体数据管理，涵盖模型配置、训练控制、贡献查看、信息维护与日志查询等关键功能。用户有自主发起训练任务、跟踪训练动态、下载训练结果的权限，并可对贡献表现进行查询。同时系统对用户的权限加以约束，保障其仅具备操作自身相关数据的权限，借此在强化操作灵活性的同时稳固系统整体的安全与公正。用户端功能设计如图 5-4 所示。

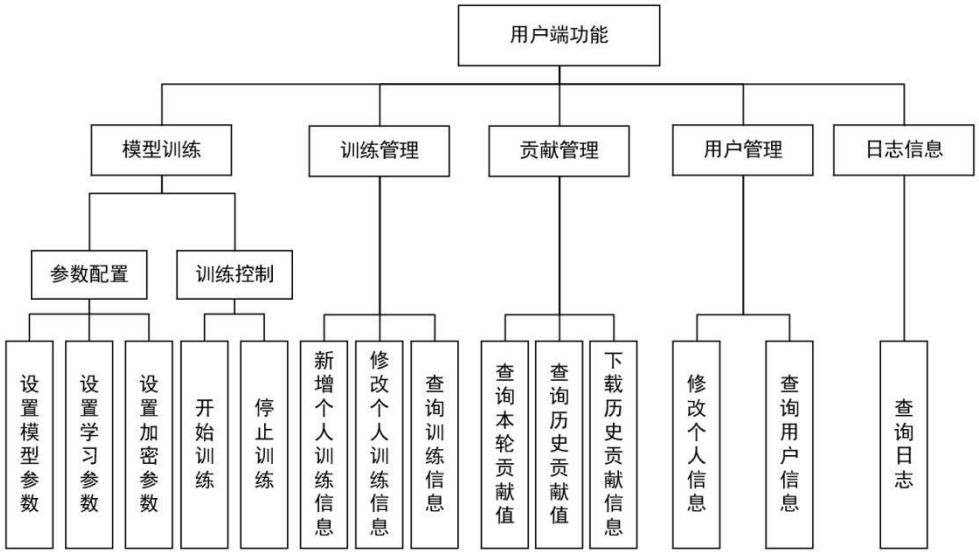


图 5-4 用户端功能设计

5.4.3 系统流程设计

本系统的核心流程围绕模型训练、训练管理、贡献管理、用户管理和日志信息五大模块展开。主要包括模型训练初始化、训练任务执行、结果存储与查询，

保障用户顺利完成训练任务并对相关数据进行统一管理。

在训练启动前，用户需完成参数配置，包括算法类型、模型选择、数据集、训练轮数、批量大小、学习率等，随后系统生成训练任务并存入数据库，便于后续管理。

训练执行过程中，系统根据配置自动开展模型训练，实时呈现训练进度，并记录训练轮次、损失值与准确率等中间结果。用户可实时查看训练状态，并在需要时中止训练。任务完成后，系统将模型结果及相关信息存储，并提供可视化反馈，方便用户分析模型性能。

训练结束后，训练记录将同步存入训练管理模块，用户可按条件筛选历史记录，普通用户可修改自身记录，管理员则具备删除权限，保障数据有序管理。

系统同时会计算客户端贡献值并更新至贡献管理模块，支持本轮与历史贡献排行查看、数据查询与排序。用户可按训练名称、编号或轮次检索贡献数据，管理员可统一管理与评估各客户端贡献，提升系统公平性与可追溯性。

此外，系统将训练过程中的关键信息写入日志管理模块，用户可查看训练日志详情以辅助调试与分析，管理员可删除无效日志，提升系统存储效率与运行稳定性。

5.5 系统数据库设计

为支持系统的训练流程管理、参数配置、用户权限控制以及贡献度评估等功能，本文设计了结构清晰、功能划分明确的数据库架构。下面主要对六张核心数据表进行介绍，分别是：用户信息表、训练信息表、模型参数表、联邦学习参数表、加密参数表与贡献信息表。

（1）用户信息表

该表用于存储系统中所有用户的基础信息，包括用户名、联系方式、身份角色及账户状态。该表是整个系统的用户管理基础，其他功能模块如训练发起人、权限控制等都依赖于此表。用户信息表如表 5-1 所示。

表 5-1 用户信息表

序号	字段名称	字段类型	说明
1	user_name	VARCHAR(64)	主键，唯一标识用户
2	role	VARCHAR(20)	用户/管理员
3	phone	VARCHAR(20)	联系电话
4	email	VARCHAR(100)	电子邮箱
5	status	VARCHAR(10)	启用/停用

（2）训练信息表

该表记录每一次发起的训练任务，包括任务标识、发起人、使用算法、所选模型和数据集、最终准确率及任务创建时间等，是系统中训练管理模块的核心数据表。训练信息表如表 5-2 所示。

表 5-2 训练信息表

序号	字段名称	字段类型	说明
1	training_id	INT	主键，唯一标识训练任务
2	training_name	VARCHAR(100)	训练任务名称
3	initiator	VARCHAR(64)	发起人，关联用户表
4	algorithm	VARCHAR(100)	联邦学习算法名称
5	start_time	DATETIME	训练任务的发起时间
6	accuracy	FLOAT	最终模型准确率
7	model_type	VARCHAR(100)	使用模型类型
8	dataset_type	VARCHAR(100)	使用数据集类型
9	remarks	TEXT	备注

（3）模型参数表

该表记录每个训练任务对应的模型参数配置，如全局训练次数、本地训练次数、学习率、动量等，它不仅支撑训练任务的初始化执行，也为后续模型效果分析与参数调优提供了可追溯的依据。模型参数表如表 5-3 所示。

表 5-3 模型参数表

序号	字段名称	字段类型	说明
1	training_id	INT	外键，关联训练任务
2	global_epochs	INT	全局训练次数
3	local_epochs	INT	本地训练次数
4	batch_size	INT	批量大小
5	learning_rate	FLOAT	学习率
6	momentum	FLOAT	动量
7	lambda_value	FLOAT	λ 值

（4）联邦学习参数表

该表存储与联邦训练策略相关的配置参数，包括客户端总数、每轮参与数量、领导人和委员会数量、压缩率等，它为系统执行联邦训练提供策略依据，影响节点选举和整体协作效率，是控制训练流程与结构的关键配置表。联邦学习参数表如表 5-4 所示。

表 5-4 联邦学习参数表

序号	字段名称	字段类型	说明
1	training_id	INT	外键，关联训练任务
2	algorithm_type	VARCHAR(100)	使用算法类型
3	total_clients	INT	总客户端数
4	per_round_clients	INT	每轮参与客户端数
5	num_leaders	INT	领导人数
6	num_verifiers	INT	验证委员会数
7	num_evaluators	INT	贡献评估委员会数
8	compression_rate	FLOAT	压缩率
9	a_value	FLOAT	α 值

（5）加密参数表

该表用于记录训练过程中涉及的加密参数配置，如所选加密算法类型、密钥长度和门限值等，为支持加密计算的模块提供参数依据。加密参数表如表 5-5 所示。

表 5-5 加密参数表

序号	字段名称	字段类型	说明
1	training_id	INT	外键，关联训练任务
2	encryption_type	VARCHAR(50)	加密类型
3	key_length	INT	密钥大小
4	threshold	INT	门限值

（6）贡献信息表

该表用于记录每轮训练中，每个客户端的贡献度表现、领导人当选情况及贡献累计结果，是支持公平激励与安全机制的重要基础表。贡献信息表如表 5-6 所示。

表 5-6 贡献信息表

序号	字段名称	字段类型	说明
1	training_id	INT	外键，关联训练任务
2	client_id	INT	客户端编号
3	round_number	INT	第几轮数据
4	round_contribution	FLOAT	当前轮贡献值
5	total_contribution	FLOAT	历史贡献值
6	leader_count	INT	累计当选领导人次数

5.6 系统开发与实现

该系统由登录功能模块、模型训练功能模块、训练管理功能模块、贡献管理

功能模块、用户管理功能模块和日志信息功能模块六大核心功能模块组成。各功能模块相互配合，共同构建完整的联邦学习管理系统，保障用户可以顺利完成模型训练，并切实有效管理相关数据。

（1）登录模块

该模块可实现用户身份验证，保障不同用户获得匹配的权限。用户填写用户名和密码后，系统进行验证，认证成功后进入相应的功能页面，用户仅能访问与自身相关的内容，管理员则可访问所有功能模块。登录页面如图 5-5 所示。

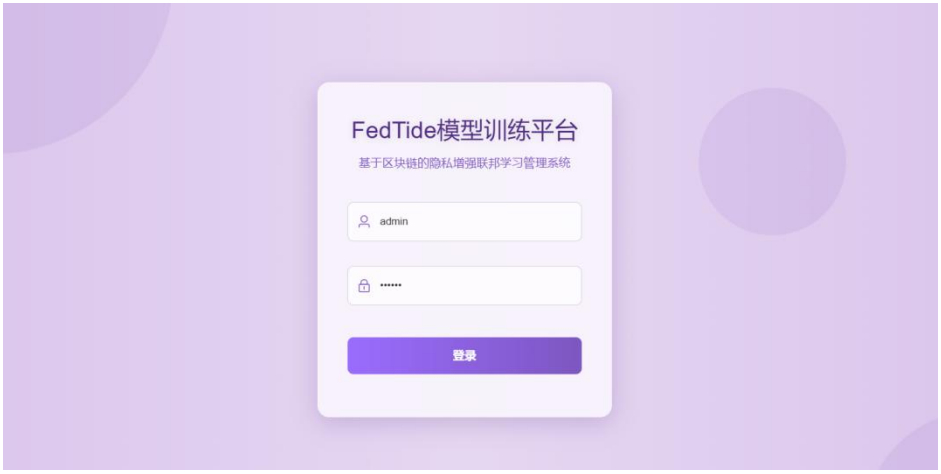


图 5-5 登录页面

（2）模型训练模块

该模块可实现模型训练，用户能够在界面中配置训练参数，包括模型类型、数据集、训练轮数、批量大小、学习率等，随后开启训练任务。在训练过程中，系统实时展示训练进度，包括当前轮次、损失值、准确率等，并提供停止训练功能。训练完成后，系统将训练结果存入训练管理模块。模型训练页面如图 5-6 所示。

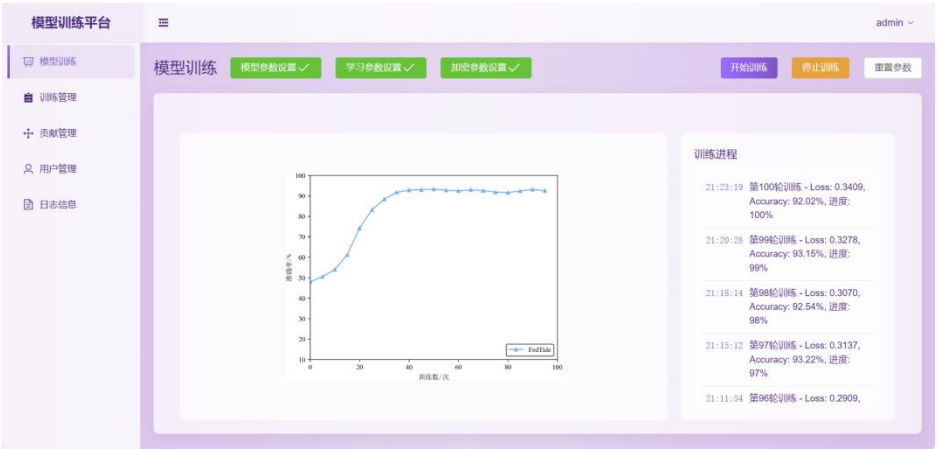


图 5-6 模型训练页面

（3）训练管理模块

该模块可实现训练记录的存储和管理，用户可通过数据集、模型类型、发起人、使用算法等条件筛选训练记录，查看训练名称、数据集、训练时间、最终准确率等详细信息并进行下载。用户可对修改训练进行修改，但不能修改或删除他人记录，管理员可实现对训练记录的删除。训练管理页面如图 5-7 所示。

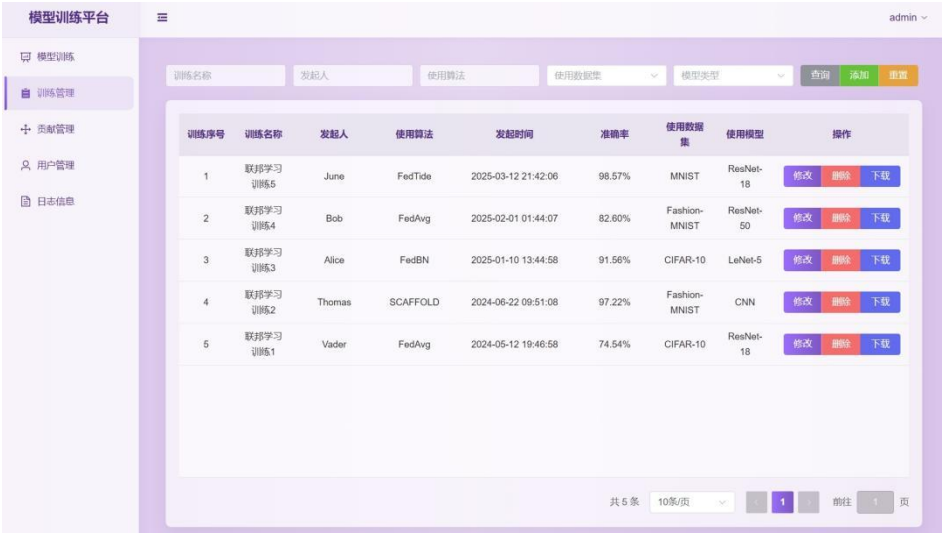


图 5-7 训练管理页面

（4）贡献管理模块

该模块可实现客户端贡献情况的展示和管理。在训练完成之后，用户可基于训练名称、训练编号或训练轮次查询贡献情况，并自行选择贡献值的升序排序或降序排序，以及当前轮次的贡献值排行和历史贡献值排行。如有需要进行后续的分析，用户也可以进行贡献数据的下载并调整贡献评估策略，管理员则可实现对贡献数据的全局管理。贡献管理页面如图 5-8 所示。

图 5-8 贡献管理页面

（5）用户管理模块

该模块可实现用户信息及权限的管理。用户可以修改个人信息（如邮箱、电话、密码等），但不能修改用户名。管理员可以添加、修改和删除用户，并对用户角色开展分配工作，以保障系统的正常管理。用户管理页面如图 5-9 所示。

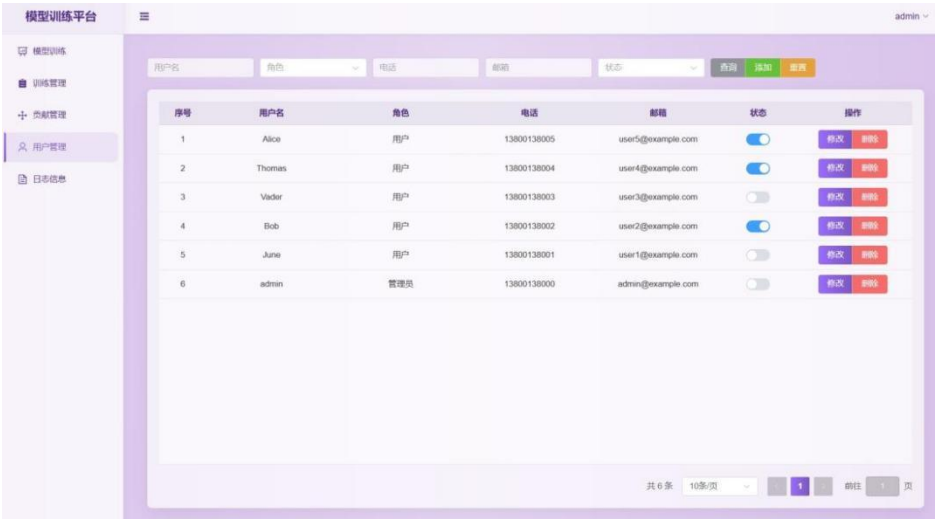


图 5-9 用户管理页面

（6）日志信息模块

该模块用于记录系统运行过程中的关键信息，保障用户可追溯系统的历史操作。日志信息主要包括训练任务的执行状态、用户操作日志和错误日志，用户可以查询所有日志，并查看训练过程的详细记录，以便在训练失败或效果不佳时进行分析和调整。但用户不能删除日志，而管理员可以删除日志，保证存储空间得到合理利用。日志信息页面如图 5-10 所示，日志详情页面如图 5-11 所示。

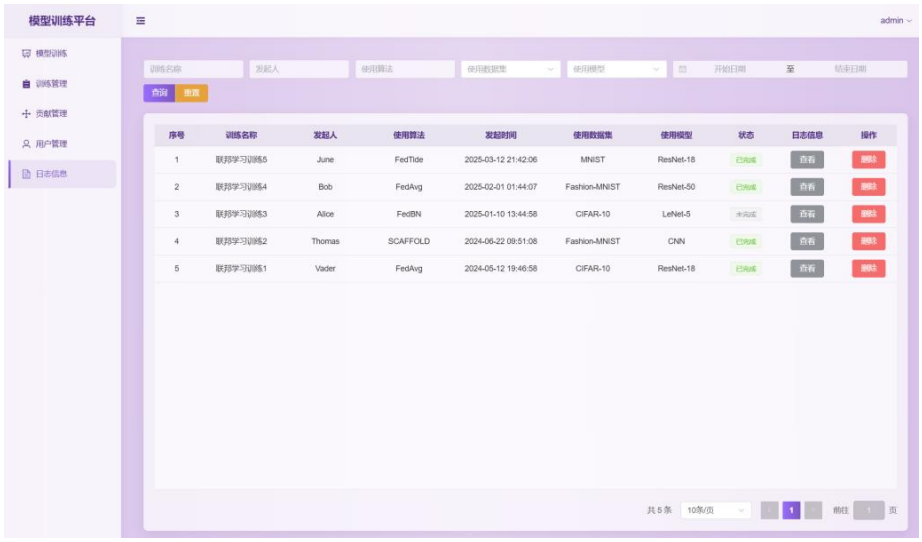


图 5-10 日志信息页面

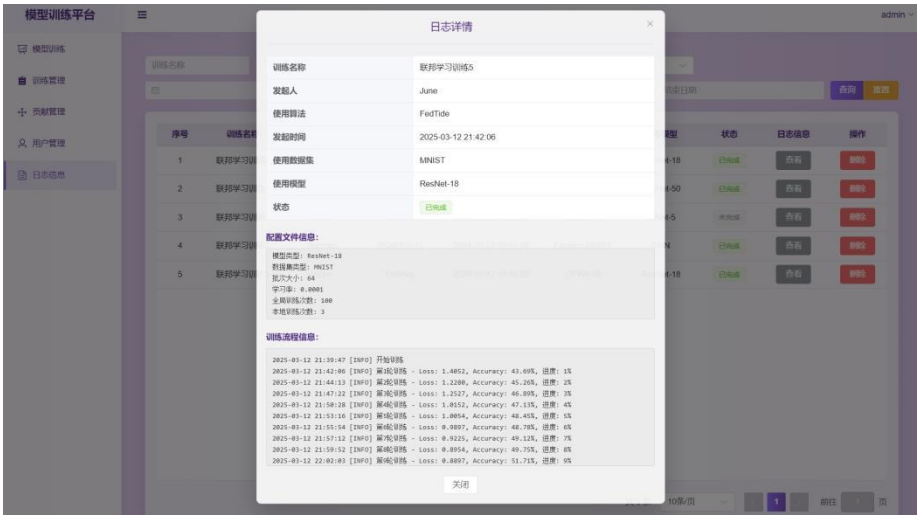


图 5-11 日志详情页面

5.7 系统测试

软件测试是验证软件系统是否符合设计要求的过程，目的在于找出系统中的错误和缺陷，保证其稳定性与可靠性。本章主要围绕功能测试和性能测试这两个方面开展分析，功能测试检查系统各模块是否正常达成目标，性能测试则评估系统在不同负载条件下的响应效率和稳定性。

5.7.1 系统功能测试

基于前文中对系统功能的设计，本系统的功能主要分为模型训练管理模块、训练记录管理模块、贡献管理模块、用户管理模块和日志管理模块。在对系统功能进行测试时，为了更好地测试相关功能是否正常，需分别针对这五部分功能设计不同的测试用例开展测试。在测试过程中，各个测试用例均会明确测试步骤及预期结果，并通过测试结果判断系统功能是否实现预期要求。测试工作不仅验证了系统功能的正确性，也为后续系统优化与稳定性提升提供了数据支撑。

（1）模型训练模块测试

模型训练模块可实现训练模型，包括训练参数配置、训练启动、训练状态监控等。用户可以在该模块中选择想要的训练算法，并在训练管理界面查询使用的算法信息。测试着重于参数的设置、训练的开展及结果的保存等方面。该模块作为联邦学习流程的起点，直接影响后续贡献评估与模型性能表现。模型训练模块测试如表 5-7 所示。

表 5-7 模型训练模块测试

测试项目	测试步骤	预期结果	测试结果
创建训练任务	输入配置参数，点击创建	配置参数成功创建	符合预期
训练任务启动	配置好参数后，点击开始训练按钮	训练任务正常启动	符合预期
训练进度监控	观察训练任务页面	进度条正常显示当前轮数、损失值、准确率等	符合预期
停止训练	在训练过程中，点击停止训练按钮	训练停止，状态更新为“已停止”	符合预期
训练结果显示	训练完成后，观察训练任务页面	训练图片正常显示	符合预期
训练记录存储	训练完成后，检查训练记录是否正常存储	训练记录正确保存	符合预期

（2）训练管理模块测试

训练管理模块可实现存储和管理用户的训练历史信息，用户可凭借特定条件（如数据集、模型类型、发起人、使用算法等）对训练记录进行查询。该模块不仅支持对训练记录的分类查看与动态维护，也为后续贡献评估与模型效果分析提供了数据基础。训练管理模块测试如表 5-8 所示。

表 5-8 训练管理模块测试

测试项目	测试步骤	预期结果	测试结果
训练记录查询	输入训练名称或编号，点击查询按钮	训练记录正确显示	符合预期
按关键信息筛选训练记录	选择算法或训练名称等进行筛选	仅显示匹配算法的训练记录	符合预期
训练记录添加	修改训练名称或备注	训练记录添加成功	符合预期
训练记录删除	选择某条训练记录，点击删除按钮	训练记录删除成功	符合预期
训练记录修改	选择某条训练记录，点击修改按钮	训练记录修改成功	符合预期
训练记录下载	下载某条训练记录，点击下载按钮	训练记录下载成功	符合预期
训练记录查询重置	点击重置按钮	输入框内容清空	符合预期

（3）贡献管理模块测试

贡献管理模块可展示客户端在每轮训练中的贡献情况，并支持多种查询及排序方式。用户可根据训练名称、训练编号或轮次筛选贡献数据，系统还支持历史贡献值下载与查询重置操作，便于进一步分析与归档。该模块在保障评估过程透明性的同时，也为后续激励分配与角色选举提供了可信的依据。贡献管理模块测试如表 5-9 所示。

表 5-9 贡献管理模块测试

测试项目	测试步骤	预期结果	测试结果
贡献值查询	选择训练名称、编号或轮次，点击查询	对应训练的贡献数据正确显示	符合预期
贡献值排序	选择按总贡献值或本轮贡献值排序	排序结果符合要求	符合预期
贡献数据下载	点击下载按钮	贡献数据表格成功导出	符合预期
贡献查询重置	点击重置按钮	输入框内容清空	符合预期

（4）用户管理模块测试

用户管理模块可实现系统用户的管理和权限控制，并支持所有用户的查询。用户管理模块测试如表 5-10 所示。

表 5-10 用户管理模块测试

测试项目	测试步骤	预期结果	测试结果
用户查询	输入用户名或选择角色等信息，点击查询	用户查询成功	符合预期
用户添加	输入用户名、密码、角色等信息，点击确定	账号创建成功	符合预期
用户登录	输入正确的账号信息，点击登录	进入系统主页	符合预期
用户信息修改	修改邮箱、电话等信息	信息更新成功	符合预期
账号删除	管理员删除某个用户	该用户从系统中移除	符合预期

（5）日志信息模块测试

日志信息模块可记录系统的操作日志。日志信息模块测试如表 5-11 所示。

表 5-11 日志信息模块测试

测试项目	测试步骤	预期结果	测试结果
日志查询	按日期、用户、操作类型筛选日志	结果符合筛选条件	符合预期
按算法筛选日志	选择某个算法进行筛选	仅显示匹配的日志记录	符合预期
日志删除	管理员选择日志并点击删除	日志删除成功	符合预期

5.7.2 系统性能测试

在系统测试过程中，性能测试同样是不可或缺的重要环节。即便系统的各项功能在常规条件下能够满足业务需求，但在高负载或极端并发场景下，仍可能出现响应延迟或资源瓶颈等问题，从而影响整体系统的稳定性与用户体验。因此，有必要对系统的关键模块在高并发场景下的响应能力和处理效率进行深入测试。

本文提出的系统包含模型训练、参数配置、加密处理、贡献评估、角色选举

等多个子模块，系统运行阶段涉及大量数据的上传、查询以及计算操作，尤其在多客户端同步进行并发训练、贡献度估算与加密通信等任务密集场景里，较高要求被施加于后端服务的计算与网络处理能力。因此本文借助 JMeter 工具开展系统性能表现的测试评估，其能够用于模拟不同并发强度状况下用户对系统的访问，测定不同线程数情况下各类请求的响应时间与系统承载能力。就本系统而言，性能测试主要聚焦于两个核心环节：模型训练任务的发起与处理过程，以及贡献管理模块中对多轮数据的查询响应效率。其中，模型训练响应时间指从用户提交训练任务到各客户端开始本地模型训练的启动延迟；贡献查询响应时间指在训练任务持续过程中，系统对特定训练编号与轮次下贡献值记录的查询速度。上述两项性能指标均与系统的处理能力密切相关，对其性能表现进行评估尤为关键。

多线程压力测试结果表明，系统在并发用户数量较情况下仍能保持稳定的响应时间，模型训练任务的平均启动延迟控制在可接受范围内，贡献查询响应速度基本稳定，说明系统具备良好的负载处理能力和并发执行能力。模型训练并发测试和贡献查询并发测试如表 5-12 和表 5-13 所示。

表 5-12 模型训练并发测试

并发线程数	平均响应时间	异常率	测试结果
10	小于 30s	0.00%	符合预期
30	小于 40s	0.00%	符合预期
60	小于 70s	0.00%	符合预期
100	小于 80s	0.00%	符合预期

表 5-13 贡献查询并发测试

并发线程数	平均响应时间	异常率	测试结果
10	小于 5s	0.00%	符合预期
30	小于 15s	0.00%	符合预期
60	小于 20s	0.00%	符合预期
100	小于 25s	0.00%	符合预期

5.8 本章小结

本章完成了系统的整体设计与实现，包括功能模块划分、数据库结构构建与核心功能开发，并结合功能测试与性能测试验证了系统在正确性、稳定性与运行效率方面的综合表现。此外，系统各模块围绕联邦学习流程紧密衔接，有效支撑了前述算法在实际环境中的部署落地。

第六章 总结与展望

6.1 总结

本文围绕区块链联邦学习的公平性、通信效率与隐私保护三大核心挑战，提出了一种基于联盟链的去中心化联邦学习框架，并针对梯度压缩与贡献评估设计了高效安全的配套机制，构建完整的贡献驱动动态协作流程。本文的研究工作主要体现在以下三方面：

首先，针对传统联邦学习存在的中心服务器单点信任与缺乏透明审计的问题，本文设计了基于联盟链的联邦学习框架。框架通过领导人节点选举、贡献评估委员会协作以及贡献排名上链等机制，实现去中心化的模型聚合与历史贡献可信存证，从系统层面增强了联邦学习过程的透明性和抗毁性。

其次，为了降低区块链联邦学习的通信负担，本文提出了动态误差感知的梯度压缩机制。该机制基于 Rand-k 稀疏化策略，结合误差反馈与动态选择概率，在充分压缩上传梯度的同时，实现对重要信息的长期补偿，有效平衡通信开销与模型收敛性能。此外，压缩梯度还结合 Paillier 加密技术，实现梯度的加密传输与安全聚合，为框架整体的隐私保护提供基础支撑。

最后，为解决传统贡献评估依赖明文梯度、难以兼顾隐私保护与评估准确性的难题，本文提出基于长期公平的历史贡献评估机制。该机制以加密余弦相似度为核心度量指标，从梯度方向贡献角度评估客户端历史贡献，并结合贡献衰减与冷却轮策略动态调整节点选举与激励分配，实现全流程透明可追溯的贡献管理与激励反馈。

6.2 展望

面向未来，本文提出的基于联盟链的去中心化联邦学习方案在通信效率、隐私保护与贡献评估透明性方面取得了初步成果，但仍存在以下值得进一步探索的方向：

一方面，关于梯度压缩与模型精度之间的平衡仍需深入研究。本文采用 Rand-k 作为核心压缩手段，虽然其具有较高的随机均衡性与隐私增强效果，但在极端稀疏率下对模型精度仍存在不可忽视的损失。未来可以结合深度学习中的注意力机制，引入梯度重要性动态预测与自适应压缩策略，在提升压缩率的同时进一步降低精度损失。

另一方面，隐私保护机制与贡献评估过程的协同增强亦是关键问题。目前的贡献评估依赖 Paillier 加密与余弦相似度的组合策略，虽然兼顾了隐私与公平，但在多轮加密计算与解密反馈过程中仍存在潜在泄漏风险。未来可以探索同态加密与零知识证明的联合验证机制，进一步提升贡献评估的安全等级，并降低评估过程中对历史数据的强依赖性。

此外，系统在激励机制与共识机制方面也具备进一步拓展空间。本文的激励机制以历史贡献为核心指标，而未来随着联邦学习应用场景的丰富化，不同业务需求可能对贡献评估标准产生差异化诉求。未来可以基于联盟链扩展智能合约能力，构建更具弹性与可扩展的多角色协作与激励体系。同时，如需支持大规模节点协作，可考虑引入分层共识或替代 PBFT 的高可扩展性方案，以减缓共识通信压力，提升系统在复杂环境下的运行效率与响应能力。

参考文献

[1] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[J]. Advances in Neural Information Processing Systems, 2012, 25.

[2] European Parliament. General data protection regulation[S]. Official Journal of the European Union, 2018.

[3] European Commission. Regulation on artificial intelligence[S]. Brussels: EU Publications, 2023.

[4] Li L, Fan Y, Tse M, et al. A review of applications in federated learning[J]. Computers & Industrial Engineering, 2020, 149: 106854.

[5] Zheng F, Li K, Tian J, et al. A vertical federated learning method for interpretable scorecard and its application in credit scoring[DB/OL]. (2020-09-14)[2025-03-21]. https://arxiv.org/abs/2009.06218.

[6] Pati S, Baid U, Edwards B, et al. Federated learning enables big data for rare cancer boundary detection[J]. Nature Communications, 2022, 13(1): 7346.

[7] Guo S, Zeng D, Dong S. Pedagogical data analysis via federated learning toward education 4.0[J]. American Journal of Education and Information Technology, 2020, 4(2): 56-65.

[8] Zhang C, Xie Y, Bai H, et al. A survey on federated learning[J]. Knowledge-Based Systems, 2021, 216: 106775.

[9] Zhang Y, Zeng D, Luo J, et al. A survey of trustworthy federated learning: issues, solutions, and challenges[J]. ACM Transactions on Intelligent Systems and Technology, 2024, 15(6): 1-47.

[10] Kalapaaking A P, Khalil I, Yi X, et al. Auditable and verifiable federated learning based on blockchain-enabled decentralization[J]. IEEE Transactions on Neural Networks and Learning Systems, 2024, 65(S2): 242322.

[11] Chen L, Zhao D, Tao L, et al. A credible and fair federated learning framework based on blockchain[J]. IEEE Transactions on Artificial Intelligence, 2024, 6(2): 301-316.

[12] Qu Y, Uddin M P, Gan C, et al. Blockchain-enabled federated learning: a survey[J]. ACM Computing Surveys, 2022, 55(4): 1-35.

[13] Liang Y, Li Y, Shin B S. Auditable federated learning with byzantine robustness[J]. IEEE Transactions on Computational Social Systems, 2023, 11: 8191-8203..

[14] Qammar A, Karim A, Ning H, et al. Securing federated learning with blockchain: a systematic literature review[J]. Artificial Intelligence Review, 2023, 56(5): 3951-3985.

[15] Liu J, Chen C, Li Y, et al. Enhancing trust and privacy in distributed networks: a comprehensive survey on blockchain-based federated learning[J]. Knowledge and Information Systems, 2024, 66(8): 4377-4403.

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

[17] Tang H, Gan S, Awan A A, et al. 1-Bit adam: Communication efficient large-scale training with adam's convergence speed[C]//International Conference on Machine Learning. PMLR, 2021: 10118-10129.

[18] Wang J, Yuan B, Rimanic L, et al. Fine-tuning language models over slow networks using activation quantization with guarantees[J]. Advances in Neural Information Processing Systems, 2022, 35: 19215-19230.

[19] Li S, Qi Q, Wang J, et al. GGS: General gradient sparsification for federated learning in edge computing[C]//ICC 2020-2020 IEEE International Conference on Communications (ICC). IEEE, 2020: 1-7.

[20] Tang Z, Shi S, Li B, et al. GossipFL: A decentralized federated learning framework with sparsified and adaptive communication[J]. IEEE Transactions on Parallel and Distributed Systems, 2022, 34(3): 909-922.

[21] Wang B, Fang J, Li H, et al. Communication-efficient federated learning: A variance-reduced stochastic approach with adaptive sparsification[J]. IEEE Transactions on Signal Processing, 2023, 71: 3562-3576.

[22] Wen W, Xu C, Yan F, et al. Terngrad: Ternary gradients to reduce communication in distributed deep learning[J]. Advances in neural information processing systems, 2017, 30: 1510-1520.

[23] Jiang J, Fu F, Yang T, et al. Skcompress: compressing sparse and nonuniform gradient in distributed machine learning[J]. The VLDB Journal, 2020, 29(5): 945-972.

[24] Campbell A, Liu H, Woldemariam L, et al. Compressed and Sparse Models for Non-Convex Decentralized Learning[DB/OL]. (2023-10-09) [2025-03-21]. https://arxiv.org/abs/2311.05760.

[25] Rivest R L, Shamir A, Adleman L. A method for obtaining digital signatures and public-key cryptosystems[J]. Communications of the ACM, 1978, 21(2): 120-126.

[26] Shafi G, Micali S. Probabilistic encryption[J]. Journal of Computer and System Sciences, 1984, 28(2): 270-299.

[27] Paillier P. Public-key cryptosystems based on composite degree residuosity classes[C]//International conference on the theory and applications of cryptographic techniques. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999: 223-238.

[28] Young T S A, Yung M. Non-interactive cryptocomputing for nc1[C]//40th Annual Symposium on Foundations of Computer Science. New York: IEEE, 1999: 554-566..

[29] Boneh D, Goh E J, Nissim K. Evaluating 2-DNF formulas on ciphertexts[C]//Theory of Cryptography: Second Theory of Cryptography Conference, TCC 2005, Cambridge, MA, USA, February 10-12, 2005. Proceedings 2. Springer Berlin Heidelberg, 2005: 325-341.

[30] Ishai Y, Paskin A. Evaluating branching programs on encrypted data[C]//Theory of Cryptography Conference. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007: 575-594.

[31] Gentry C. Fully homomorphic encryption using ideal lattices[C]//Proceedings of the forty-first annual ACM symposium on Theory of computing. 2009: 169-178.

[32] Brakerski Z, Vaikuntanathan V. Fully homomorphic encryption from ring-LWE and security for key dependent messages[C]//Annual cryptology conference. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011: 505-524.

[33] Van Dijk M, Gentry C, Halevi S, et al. Fully homomorphic encryption over the integers[C]//Advances in Cryptology-EUROCRYPT 2010: 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques, French Riviera, May 30-June 3, 2010. Proceedings 29. Luxembourg: Springer Science & Business Media, 2010: 24-43.

[34] Yang X, Xiang S, Peng C, et al. Federated learning incentive mechanism design via Shapley value and Pareto optimality[J]. Axioms, 2023, 12(7): 636.

[35] Shi Z, Zhang L, Yao Z, et al. Fedfaim: A model performance-based fair incentive mechanism for federated learning[J]. IEEE Transactions on Big Data, 2022, 10(6): 1038-1050.

[36] Yang X, Tan W, Peng C, et al. Federated learning incentive mechanism design via enhanced shapley value method[J]. Wireless Communications and Mobile Computing, 2022, 2022(1): 9690657.

[37] Pan Z, Wang S, Li C, et al. Fedmdfg: Federated learning with multi-gradient descent and fair guidance[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2023, 37(8): 9364-9371.

[38] Wang Z, Fan X, Qi J, et al. Federated learning with fair averaging[DB/OL]. (2021-04-21)[2025-03-21]. https://arxiv.org/abs/2104.14937.

[39] Yan B, Jiang X, Chen Y, et al. AFL-CS: Asynchronous Federated Learning with Cosine Similarity-based Penalty Term and Aggregation[C]//2023 IEEE 29th International Conference on Parallel and Distributed Systems (ICPADS). IEEE, 2023: 46-53.

[40] Wu S, Zhou Y, Gao X, et al. K Asynchronous Federated Learning with Cosine Similarity Based Aggregation on Non-IID Data[C]//International Conference on Algorithms and Architectures for Parallel Processing. Singapore: Springer Nature Singapore, 2023: 434-452.

[41] Ren P, Qi K, Li J, et al. CosPer: An adaptive personalized approach for enhancing fairness and robustness of federated learning[J]. Information Sciences, 2024, 675: 120760.

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

[43] Liu D, Bai L, Yu T, et al. Towards method of horizontal federated learning: A survey[C]//2022 8th international conference on big data and information analytics (BigDIA). IEEE, 2022: 259-266.

[44] Liu Y, Kang Y, Zou T, et al. Vertical federated learning: Concepts, advances, and challenges[J]. IEEE Transactions on Knowledge and Data Engineering, 2024, 36(7): 3615-3634.

[45] Tan Y N, Tinh V P, Lam P D, et al. A transfer learning approach to breast cancer classification in a federated learning framework[J]. IEEe Access, 2023, 11: 27462-27476.

[46] Zhai S, Yang Y, Li J, et al. Research on the Application of Cryptography on the Blockchain[C]//Journal of Physics: Conference Series. IOP Publishing, 2019, 1168: 032077.

[47] Li W, Feng Y, Liu N, et al. A secure and efficient log storage and query framework based on blockchain[J]. Computer Networks, 2024, 252: 110683.

[48] Xie M, Liu J, Chen S, et al. A survey on blockchain consensus mechanism: research overview, current advances and future directions[J]. International Journal of Intelligent Computing and Cybernetics, 2023, 16(2): 314-340.

[49] Zhu Y. Security architecture and key technologies of blockchain[J]. Journal of Information Security Reserach, 2016, 2(12): 1090.

[50] Li C, Qiu W, Li X, et al. A dynamic adaptive framework for practical byzantine fault tolerance consensus protocol in the internet of things[J]. IEEE Transactions on Computers, 2024, 73(7): 1669-1682.

[51] Wahab N H A, Zhang D, Juniardi N F, et al. Advances in Consortium Chain Scalability: A Review of the Practical Byzantine Fault Tolerance Consensus Algorithm[J]. International Journal of Advanced Computer Science and Applications (IJACSA), 2024, 7.

[52] Castro M, Liskov B. Practical byzantine fault tolerance[C]//OsDI. 1999, 99(1999): 173-186.

[53] Rivest R L, Adleman L, Dertouzos M L. On data banks and privacy homomorphisms[J]. Foundations of secure computation, 1978, 4(11): 169-180.

[54] Marcolla C, Sucasas V, Manzano M, et al. Survey on fully homomorphic encryption, theory, and applications[J]. Proceedings of the IEEE, 2022, 110(10): 1572-1609.

[55] Font J M, Jansana R, Pigozzi D. A survey of abstract algebraic logic[J]. Studia Logica, 2003, 74: 13-97.

[56] Catalano D, Gennaro R, Howgrave-Graham N, et al. Paillier's cryptosystem revisited[C]//Proceedings of the 8th ACM Conference on Computer and Communications Security. 2001: 206-214.

[57] Brakerski Z. Fully homomorphic encryption without modulus switching from classical GapSVP[C]//Annual cryptology conference. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012: 868-886.

[58] Reisizadeh A, Mokhtari A, Hassani H, et al. An exact quantized decentralized gradient descent algorithm[J]. IEEE Transactions on Signal Processing, 2019, 67(19): 4934-4947.

[59] Han P, Wang S, Leung K K. Adaptive gradient sparsification for efficient federated learning: An online learning approach[C]//2020 IEEE 40th international conference on distributed computing systems (ICDCS). Singapore: IEEE, 2020: 300-310.

[60] Shahid O, Pouriyeh S, Parizi R M, et al. Communication efficiency in federated learnin g: Achievements and challenges[DB/OL]. (2021-07-23) [2025-03-21]. https://arxiv.org/abs/ 2107.10996.

[61] Krishnamoorthi R. Quantizing deep convolutional networks for efficient inference: A whitepaper[DB/OL]. (2018-06-21)[2025-03-21]. https://arxiv.org/abs/1806.08342.

[62] Sun X, Wang N, Chen C Y, et al. Ultra-low precision 4-bit training of deep neural networks[J]. Advances in Neural Information Processing Systems, 2020, 33: 1796-1807.

[63] Jacob B, Kligys S, Chen B, et al. Quantization and training of neural networks for efficient integer-arithmetic-only inference[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. Piscataway, NJ: IEEE, 2018: 2704-2713.

[64] Dong Z, Yao Z, Gholami A, et al. Hawq: Hessian aware quantization of neural networks with mixed-precision[C]//Proceedings of the IEEE/CVF international conference on computer vision. Seoul: IEEE, 2019: 293-302.

[65] Wang Z, Wen M, Xu Y, et al. Communication compression techniques in distributed deep learning: A survey[J]. Journal of Systems Architecture, 2023, 142: 102927.

[66] Cao X, Başar T, Diggavi S, et al. Communication-efficient distributed learning: An overview[J]. IEEE Journal on Selected Areas in Communications, 2023, 41(4): 851-873.

[67] Shi S, Chu X, Cheung K C, et al. Understanding top-k sparsification in distributed deep learning[DB/OL]. (2019-11-20)[2025-03-21]. https://arxiv.org/abs/1911.08772.

[68] Jiang S, Sharma P, Joshi G. Correlation aware sparsified mean estimation using random projection[J]. Advances in Neural Information Processing Systems, 2023, 36: 53892-53923.

[69] Tyurin A, Richtarik P. 2Direction: Theoretically faster distributed training with bidirectional communication compression[J]. Advances in Neural Information Processing Systems, 2023, 36: 11737-11808.

[70] Mohri M, Sivek G, Suresh A T. Agnostic federated learning[C]//International conference on machine learning. Vienna, Austria: PMLR, 2019: 4615-4625.

[71] Sun Y, Si S, Wang J, et al. A fair federated learning framework with reinforcement learning[C]//2022 International Joint Conference on Neural Networks (IJCNN). Padua, Italy: IEEE, 2022: 1-8.

[72] Tian J, Lü X, Zou R, et al. A fair resource allocation scheme in federated learning[J]. Journal of Computer Research and Development, 2022, 59(2022-06): 1240.

[73] Sarikaya Y, Ercetin O. Motivating workers in federated learning: A stackelberg game perspective[J]. IEEE Networking Letters, 2019, 2(1): 23-27.

[74] Wang G, Dang C X, Zhou Z. Measure contribution of participants in federated learning[C]//2019 IEEE international conference on big data (Big Data). Los Angeles, CA: IEEE, 2019: 2597-2604.

[75] Kang J, Xiong Z, Niyato D, et al. Reliable federated learning for mobile networks[J]. IEEE Wireless Communications, 2020, 27(2): 72-80.

[76] Cui S, Pan W, Liang J, et al. Addressing algorithmic disparity and performance inconsistency in federated learning[J]. Advances in Neural Information Processing Systems, 2021, 34: 26091-26102.

[77] Li X, Jiang M, Zhang X, et al. Fedbn: Federated learning on non-iid features via local batch normalization[DB/OL]. (2021-05-11)[2025-03-21]. https://arxiv.org/abs/1911.08772.

[78] Karimireddy S P, Kale S, Mohri M, et al. Scaffold: Stochastic controlled averaging for federated learning[C]//International conference on machine learning. Vienna, Austria: PMLR, 2020: 5132-5143.

[79] Kirszbraun M. Über die zusammenziehende und Lipschitzsche Transformationen[J]. Fundamenta Mathematicae, 1934, 22(1): 77-108.

[80] Boyd S P, Vandenberghe L. Convex optimization[M]. Cambridge university press, 2004.

[81] Shalev-Shwartz S, Singer Y, Srebro N. Pegasos: Primal estimated sub-gradient solver for svm[C]//Proceedings of the 24th international conference on Machine learning. 2007: 807-814.

[82] Horvóth S, Ho C Y, Horvath L, et al. Natural compression for distributed deep learning[C]//Mathematical and Scientific Machine Learning. PMLR, 2022: 129-141.