基于同态加密的高效安全联邦学习聚合框架

余晟兴，陈钟
（北京大学计算机学院，北京 100871）

摘 要：为了解决联邦学习数据安全以及加密后通信开销大等问题，提出了一种基于同态加密的高效安全联邦聚合框架。在联邦学习过程中，用户数据的隐私安全问题亟须解决，然而在训练过程中采用加密方案带来的计算和通信开销又会影响训练效率。在既要保护数据安全又要保证训练效率的情况下，首先，采用 Top-K 梯度选择方法对模型梯度进行筛选，减少了需要上传的梯度数量，提出适合多边缘节点的候选量化协议和安全候选索引合并算法，进一步降低通信开销、加速同态加密计算。其次，由于神经网络每层模型参数具有高斯分布的特性，对选择的模型梯度进行裁剪量化，并采用梯度无符号量化协议以加速同态加密计算。最后，实验结果表明，在联邦学习的场景下，所提框架既保证了数据隐私安全，又具有较高的准确率和高效的性能。

关键词：联邦学习；同态加密；隐私保护；量化协议
中图分类号：TN92
文献标志码：A
DOI: 10.11959/j.issn.1000−436x.2023015

Efficient secure federated learning aggregation framework based on homomorphic encryption

YU Shengxing, CHEN Zhong
School of Computer Science, Peking University, Beijing 100871, China

Abstract: In order to solve the problems of data security and communication overhead in federated learning, an efficient and secure federated aggregation framework based on homomorphic encryption was proposed. In the process of federated learning, the privacy and security issues of user data need to be solved urgently. However, the computational cost and communication overhead caused by the encryption scheme would affect the training efficiency. Firstly, in the case of protecting data security and ensuring training efficiency, the Top-K gradient selection method was used to screen model gradients, reducing the number of gradients that need to be uploaded. A candidate quantization protocol suitable for multi-edge terminals and a secure candidate index merging algorithm were proposed to further reduce communication overhead and accelerate homomorphic encryption calculations. Secondly, since model parameters of each layer of neural networks had characteristics of the Gaussian distribution, the selected model gradients were clipped and quantized, and the gradient unsigned quantization protocol was adopted to speed up the homomorphic encryption calculation. Finally, the experimental results show that in the federated learning scenario, the proposed framework can protect data privacy, and has high accuracy and efficient performance.

Keywords: federated learning, homomorphic encryption, privacy-preserving, quantization protocol

0 引言

机器学习在许多应用场景中发挥着重要的作用，如银行信用评估、医疗诊断、语音识别等，因此机器学习的扩展应用受到了学术界和工业界的广泛关注。机器学习已经逐渐改变人类的衣食住行习惯，并渗透生活的各个方面。机器学习往往需要大规模的数据来保证模型的准确率，且用户、组织

收稿日期：2022−08−17；**修回日期：**2022−11−28
通信作者：陈钟，zhongchen@pku.edu.cn

或者公司自身的算力有限，通常将数据和计算外包给云服务，针对数据合作和隐私保护需求，联邦学习（FL, federated learning）应运而生，其允许用户将个人隐私信息、组织机密信息、公司内部资料等数据留在本地进行模型训练，在云端聚合全局模型，但在交互过程中仍有隐私信息泄露的风险，一旦数据泄露，用户、组织或者公司将会遭受重大损失。由于人们对隐私数据可能遭到泄露感到担忧，相关法律法规[1]明确禁止收集和利用未授权的敏感数据。尽管在数据共享的同时满足用户敏感信息的隐私保护存在一定的困难，但由于最先进的机器学习模型需要大量的数据参与训练，不同组织或者公司对数据共享需求仍然很强烈。因此，如何让不同的数据持有者在联邦学习场景下保证用户隐私不被泄露是构建一个高质量的机器学习模型所面临的重要问题。

为了保证模型训练中用户敏感数据的隐私性和训练过程的安全性，通常采用 3 种核心的隐私保护技术，即安全多方计算（SMC, secure multi-party computation）[2]、差分隐私（DP, differential privacy）[3]和同态加密（HE, homomorphic encryption）[4]。SMC（包括混淆电路、秘密分享等）[2]允许多个边缘节点将数据以秘密分享的性质进行隐私计算，在保持该数据私有的情况下多边缘节点协作评估一个函数。DP[3]在数据被第三方交换和分析时，为数据添加适当的校准噪声以消除个人身份的歧义，并计算隐私预算来保证其计算的准确性，由于 DP 添加了噪声，其牺牲了训练/预测的准确率来达到快速计算的目的。HE[4]是一种在保护数据隐私下避免隐私泄露风险的解决方案。现有基于 HE 的隐私保护机器学习机制主要依赖于单云模型或双云模型的云计算框架，该框架已经扩展到边缘计算[5-6]。虽然单云模型[7]比双云模型更容易导致隐私泄露，但双云模型的实际应用[8-9]是基于 2 个半诚实云服务器之间不共谋的强假设。考虑到机器学习/深度学习的训练阶段涉及加密数据的大量安全计算，单云模型相比于双云模型不仅在通信上减少了云服务器之间的交互，提高了计算效率，还避免了云服务器之间的窃取攻击问题。

FL 用于从所有可用数据集中训练具有稳健性的模型。边缘节点进行本地训练后，将训练后的梯度上传至云服务器，由云服务器进行聚合更新。虽然该技术很有前景，但 Li 等[10]提出了 4 个限制 FL 技术在现实世界中大规模应用的原因，即昂贵的通信、系统异质性、统计异质性和隐私问题。本文针对通信和隐私挑战提出高效的解决方案。特别地，尽管让数据留在各边缘节点可以保障数据不被直接泄露，但是共享中间模型更新已经被证明是会泄露敏感信息[11-12]的。为了解决这个问题，联邦学习经常将模型训练和现有的 HE 或 SMC 等技术结合在一起，以确保模型训练传输过程中不会泄露任何敏感信息。

此外，SMC 技术可用于安全地聚合本地模型更新而不暴露模型参数，它可以确保隐私保护下的各个模型梯度不被泄露，然而 SMC 技术应用的主要瓶颈是额外的计算和通信成本。相比于 HE，SMC 需要双云服务器甚至多台云服务器之间相互协作完成计算，这带来了更大的通信开销。而在 HE 技术中，公私密钥对通过一个安全通道传输到所有边缘节点，每个边缘节点使用公钥加密其更新的梯度，并将密文上传到中心服务器。中心服务器聚合所有从边缘节点上传的加密梯度，并将结果分发给每一个边缘节点。边缘节点使用私钥解密聚合后的梯度，更新其本地模型，并进行下一次迭代。由于边缘节点只上传加密的更新梯度，服务器无法在外部或者数据传输过程中获得任何信息，虽然 HE 为 FL 提供了强大的隐私保障，但它复杂的密文操作（如大整数乘法和指数）的计算成本是十分昂贵的，甚至超过 80%的训练迭代时间花费在同态加密/解密上[13]，HE 的加密和通信开销已成为其应用在联邦学习上的主要障碍。针对此问题，本文设计了高效的同态加密安全联邦聚合框架（ESFL）。

本文主要贡献如下。

1) 本文采用 Top-K 梯度选择算法对需要上传的模型梯度进行筛选，压缩需要上传梯度的数量，降低了边缘节点之间同步梯度的通信开销。目前的方案[14]将秘密分享与 Top-K 梯度选择相结合，存在暴露聚合更新后梯度索引值的安全问题，本文通过同态加密既保证了各边缘节点更新梯度的安全性，也保证了服务器聚合后无法获得模型隐私。

2) 本文采用批量量化和编码的加密技术来解决加密和通信瓶颈问题。边缘节点将 Top-K 选择的梯度索引值和需要更新的梯度进行批量量化并分别编码为一个长整数进行加密上传，相比于对数据逐个进行加密上传的方法，该方法大幅

降低了时间开销。目前，Zhang 等[13]采用的量化加密方案虽然解决了同态加密计算开销问题，但是其无法适应多边缘节点的场景，本文在其方案上进行了改进，解决了该方案多边缘节点时的数据溢出问题。

3) 本文提出了高效的同态加密安全联邦聚合框架 ESFL，解决了基于联邦学习的梯度聚合方案存在的隐私泄露以及通信开销大的问题。ESFL 基于 Top-K 梯度选择算法对梯度进行一次筛选，减少上传梯度的数量；将梯度索引进行量化压缩编码成大整数后加密，大幅降低了同态加密/解密以及服务器隐私计算的时间；最后裁剪量化需要更新的梯度，并加密量化后所编码的大整数，进一步提升了计算的效率。所提框架在联邦学习的多个步骤中都进行了效率优化，既保证了数据的隐私安全，又突破了同态加密带来的效率瓶颈。

1 相关工作

1.1 Top-K 梯度选择

Strom[15]通过将绝对值超过阈值的梯度元素设置为 1 来选择需要上传的梯度，并采用 1 bit 量化以及梯度残差补偿的方法，大幅度减小带宽。Dryden 等[16]针对 Strom 阈值选择困难的缺陷，提出自适应量化的方法，使用一个固定的比例，通过确定正阈值和负阈值来决定每个最小批处理要发送的梯度更新的比例。Aji 等[17]通过移除一定比例的最小梯度绝对值来稀疏梯度更新。然而此方案与 Dryden 等[16]方案略有不同，其使用的是基于绝对值的单一阈值。Alistarh 等[18]提出的方案不再需要计算阈值来选择梯度，在分析假设下，通过局部误差校正，为压缩后的梯度采用随机梯度下降（SGD, stochastic gradient descent）提供了收敛保证，选择前 Top-K 个变化幅度大的梯度作为需要更新的梯度。

1.2 联邦学习安全聚合

为了解决联邦学习隐私安全问题，Bonawitz 等[19]提出了基于半诚实模型的安全、高效和稳健的聚合协议，其采用 Shamir 秘密分享技术[20]实现安全聚合协议。Niu 等[21]提出了一种低通信、低计算开销的联邦学习安全聚合方法，在每个边缘节点的子模型中采用局部差分隐私保护隐私数据，并设计了一个基于布隆过滤器[22]和安全聚合的高效可扩展私有集合联合协议。Dong 等[23]基于 TernGrad 提出了通过压缩模型参数来实现高效联邦学习的聚合方法，并提出了使用同态加密和阈值秘密分享技术来实现安全聚合方案，但该方案的通信和计算开销很大，无法实现高效的性能。Zhang 等[13]提出了基于同态加密的安全聚合方法，通过梯度裁剪量化以及拼接操作，在一定程度上解决了同态加密技术带来的通信和计算开销巨大的问题。

2 理论知识

2.1 同态加密技术

同态加密技术 paillier 密码系统是一种加性同态概率非对称加密方案[23]。设 E_{pk} 为(N,g) 公钥 pk 的加密函数，其中，N 是 2 个大素数的乘积，$g\in Z_{N\times N}^{*}$。同时，设 D_{sk} 为具有密钥 sk 的解密函数。给定 $a,b\in Z_N$，为了简化表示，将同态加密后的数表示为 $[\![\cdot]\!]$，其数组形式表示为 $\langle\cdot\rangle$。paillier 加密方案具有以下性质：为了计算 2 个密文 $[\![a]\!]$ 和 $[\![b]\!]$ 的和，可以通过密文之间的乘法，即 $[\![z]\!]=[\![a]\!][\![b]\!]=[\![a+b]\!]$ 来实现；为了进行明文与密文的乘法运算，即数字 $[\![a]\!]$ 与密文 $[\![b]\!]$ 的乘积，可以通过 $[\![z]\!]=[\![ab]\!]=[\![a]\!]^{b} \bmod N^2$ 来实现。

2.2 dACIQ 裁剪量化协议

由于最先进的削波技术即对称量化算法（ACIQ）[25]无法进行非对称量化，且需要获取模型参数的均值，无法避免需要上传所有模型参数，Zhang 等[13]提出的裁剪方案即基于非中心化数据的分析模型（dACIQ）仅需要上传每层模型梯度的极值并计算出阈值。此外，来自不同层的数据具有不同的分布[26]，需要单独量化[26-27]每一层梯度，且先前的工作表明来自同一层梯度的分布接近高斯的钟形分布[28-29]，通过该性质可以考虑将梯度有效地压缩到某个高斯分布上[26-27]。

假设 dACIQ 计算出的裁剪阈值为 $\alpha\in[0,2^{64}-1]$，梯度服从高斯分布 $X\sim N(0,\sigma^2)$，则有如图 1 所示的典型的层梯度分布。按照阈值裁剪梯度，将会产生累积噪声，包括舍入噪声和裁剪噪声，其中，舍入噪声是指在阈值范围内取整所产生的误差，裁剪噪声是指超过阈值裁剪所产生的误差。特别地，为了衡量裁剪噪声，使用 δ_c 来表示。

$$\delta_c=\int_{-\infty}^{-\alpha} f(x)(x+\alpha)^2\mathrm{d}x+\int_{\alpha}^{+\infty} f(x)(x-\alpha)^2\mathrm{d}x \quad (1)$$

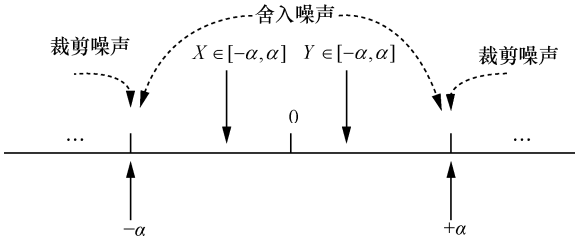


图 1　典型的层梯度分布

累计误差中的舍入噪声 δ_r 表示为

$$\delta_r = \sum_{i=0}^{2^r+3} \int_{q_i}^{q_{i+1}} f(x) \cdot \left[(x+q_i)^2 \left(\frac{q_{i+1}-x}{\Delta} \right) + (X-q_{i+1})^2 \left(\frac{x-q_i}{\Delta} \right) \right] \mathrm{d}x \quad (2)$$

根据 δ_c 和 δ_r 可知，累计误差表示为

$$\mathrm{Err} = \delta_c + \delta_r \approx \frac{\alpha^2+\sigma^2}{2} \left[1 - \mathrm{erf}\left(\frac{\alpha}{\sqrt{2}\sigma} \right) \right] - \frac{\alpha\sigma \mathrm{e}^{-\frac{\alpha^2}{2\sigma^2}}}{\sqrt{2\pi}} + \frac{2\alpha^2(2^r-2)}{3 \cdot 2^{3r}} \quad (3)$$

其中，r 为量化宽度，q_i 为第 i 个量化水平，erf 是误差函数，为近似密度函数，即分段线性函数，Δ 为最小量化步长。从式(3)可知，只要得到 σ，即可推导出使 Err 最小的阈值 α，并将其作为裁剪阈值。

一般来说，由于每层的梯度具有高斯分布特性，梯度重新拟合到高斯分布需要确定高斯分布中的 σ 和 μ。传统拟合高斯分布的 σ 和 μ 是采用极大似然估计和贝叶斯推理得到的，且需要的信息包括观测集大小、观测值以及观测值平方和。由于神经网络的每层梯度数量可能有数十万甚至上百万个，如果通过传统的拟合方法得到参数 σ 和 μ，其时间和通信成本是非常昂贵的。因此采用 Banner 等[25,30]提出的一种简单而高效的高斯拟合方法计算出 σ，其假设高斯随机变量最大值和最小值的期望有界为

$$0.23\sigma \leqslant \frac{\mathrm{E}[\max(x^{(d)} - \mu^{(d)})]}{\sqrt{\ln(n)}} \leqslant \sqrt{2}\sigma \quad (4)$$

$$-0.23\sigma \leqslant \frac{\mathrm{E}[\min(x^{(d)} - \mu^{(d)})]}{\sqrt{\ln(n)}} \leqslant -\sqrt{2}\sigma \quad (5)$$

其中，x^d 是输入 x 的第 d 个元素，μ^d 是 x^d 的期望值，n 是批处理大小。根据式(4)和式(5)可知，该方法只需要观测集的大小及其极大值和极小值，计算和通信开销最小，且文献[30-31]的实验表明，这种拟合方法对模型的准确性不产生影响。由于 ESFL 存在多个边缘节点，各边缘节点梯度的边界不一致，可以通过提前放缩将梯度裁剪到 $[-\alpha,\alpha]$，采用对称边界可以有效降低计算量。

3　问题定义

3.1　系统模型

系统模型如图 2 所示，包括密钥生成中心（KGC, key generation center）、云平台（CP, cloud platform）和边缘节点（EN, edge node）3 类实体。假设系统中包含 N 个边缘节点，实体之间的通信是与安全通道同步的。每个实体的具体作用如下。

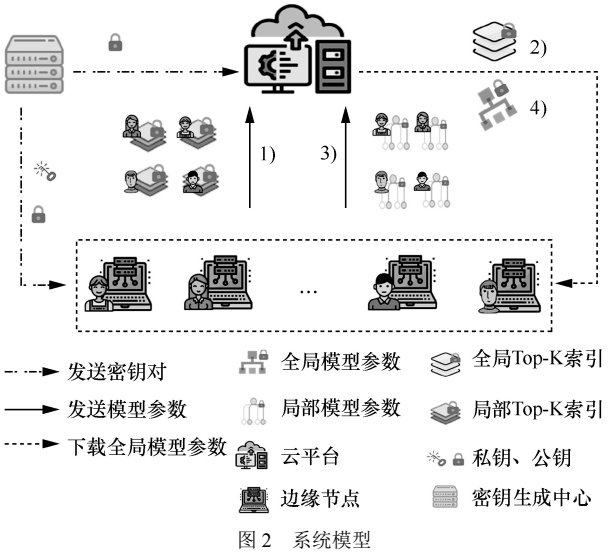


图 2　系统模型

1) KGC。对于边缘节点，KGC 完全可信，且为系统生成、管理和分发密钥。

2) CP。CP 为边缘节点提供无限算力和存储容量，主要为 N 个 EN 的隐私数据提供隐私计算服务。

3) EN。边缘节点是存储空间和计算能力有限的个人/组织，用于存储有限的用户敏感数据。在训练阶段，EN 愿意与其他 EN 协作共同构建全局模型，而不直接提供各自的隐私数据。

3.2　威胁模型

从对抗的角度来看，根据系统中实体 CP 和 EN 访问的信息来考虑系统可能面临的威胁。具体威胁如下。

来自实体的威胁。假设 KGC 的密钥分发是可信

的，且 N 个 EN 和 CP 被认为是诚实地遵循特定协议但试图从加密数据中获取额外信息的诚实且好奇的实体。在实际执行过程中，假设 CP 和任意 EN 合谋，那么将暴露 EN 的隐私信息，因此，CP 和 EN 不值得与其他实体勾结以免个人隐私泄露，综上所述，假设 CP 和 EN 之间不存在相互勾结。

来自外部对手的威胁。假设外部对手能够从 EN 和 CP 之间的通信频道窃听传输的信息，并且还有一个对手可以破坏 EN 或 CP。

3.3 设计目标

ESFL 系统旨在实现轻量级计算的隐私保护机器学习框架。设计目标如下。

安全。由于每个 EN 上所构建的模型都包含模型隐私信息，应保证从 EN 发送出去的数据的隐私性；在各 EN 与 CP 联合训练过程中，应保证数据在交互以及中间计算时的隐私安全。

高效。ESFL 系统应保证 EN 与 CP 联合训练过程中的高效性，同时实现 N 个 EN 与 CP 之间较低的通信开销，有效降低 CP 和 EN 的通信负载。

准确。ESFL 系统应保持可靠、准确的模型训练，保证较高的模型表达，为每个 EN 提供准确的预测结果。

4 高效联邦安全聚合框架

4.1 框架描述

本节描述了 ESFL 系统的设计过程，这一过程包括 4 个主要阶段。

1) 密钥生成。为了提供隐私保护，KGC 首先生成密钥对即公钥和私钥(pk,sk)，并将公钥 pk 发送给每个 EN 和 CP，同时私钥 sk 发送给每个 EN 用于解密数据。

2) 安全候选索引合并。为了降低通信开销，N 个 EN 和 CP 根据每个 EN 的 Top-K 选择的梯度联合更新全局模型，因此云服务器 CP 需要对各 EN 的候选索引进行合并。为了进一步降低合并开销，每个 EN 采用索引量化协议，将梯度索引值量化成 EN 数量的二进制比特位，并批量拼接成大整数，而 CP 仅对上传的密文进行同态加法，EN 最后通过解密密文并反量化确定需要上传的梯度。

3) 裁剪量化梯度。根据安全候选索引合并确定的候选梯度集合，每个 EN 上传对应的模型参数的极值和量化位宽，同时 CP 通过 dACIQ 计算出裁剪阈值 α。每个 EN 根据 α 裁剪候选梯度，量化成无符号整数，并将量化的候选梯度批量拼接成大整数。

4) ESFL 安全聚合。在各 EN 本地上，EN 构建局部模型，CP 聚合 EN 发送过来的所有局部模型，同时 EN 根据 CP 构建的全局模型更新局部模型。如图 2 所示，具体过程分为以下 4 个步骤。

步骤 1 每个 EN 首先在本地训练一定轮次模型后根据 Top-K 选择模型参数，并对候选索引量化加密，之后上传加密的候选索引到 CP。

步骤 2 CP 收集所有 EN 发送过来的加密候选索引，并执行安全候选索引合并协议，发送合并后的候选索引集合到每个 EN 上。

步骤 3 根据 CP 发送的候选索引集合，每个 EN 与 CP 交互并利用对应的模型参数计算裁剪阈值，对候选梯度进行裁剪并量化；每个 EN 根据候选梯度批量拼接成大整数，之后通过 pk 加密候选梯度并上传到 CP。

步骤 4 CP 通过安全聚合上传候选梯度得到全局模型，并将其发送到各 EN，EN 通过私钥 sk 解密，对梯度进行反量化操作，并更新局部模型参数。

4.2 Top-K 梯度选择算法

现有两倍压缩技术[31]压缩了边缘节点发送到服务器的局部模型，以及从服务器返回的全局模型。然而，重新对聚合后的全局模型进行压缩需要获得模型参数更新的明文值范围，或者需要安全、高效地执行隐私保护下的重压缩协议，现有的技术无法满足。因此，为了保护隐私和安全，仅对边缘节点发送到服务器的模型参数进行压缩，而不重新压缩服务器返回的模型参数。

Top-K 梯度选择[32]的目的是通过减少上游（从 EN 到 CP）和下游（从 CP 到 EN）交互的数据量来减少通信量，选择边缘节点模型梯度变化幅度最大的前 k 个参数作为上传的梯度，Alistarh 等[18]已证明了其收敛性。在 ESFL 中，每个 EN 在每一轮只更新各 EN 选择的大小为 k 的梯度并集，CP 和 EN 不需要传递其余模型梯度的权重，从而其余的模型梯度在整个训练过程中都是保持不变的。具体算法过程如算法 1 所示。

算法 1 梯度选择协议（TKP）

输入 上一轮模型 $\boldsymbol{G}^{\text{glob}}$，当前模型 $\boldsymbol{G}^{\text{cur}}$

输出 候选索引数组 L

1) 定义 TOPK 函数是获得降序排序后前 k 个的梯度对应的索引值；

2) $g=[0,\cdots,0]_m$ //m 为模型包含的梯度个数

3) for i=1:1: m

4) $g_i = (\boldsymbol{G}_i^{\text{glob}} - \boldsymbol{G}_i^{\text{cur}})^2$；

5) end for

6) $L = \text{TOPK}(g)$。

4.3　安全候选索引合并算法

为了降低通信开销，采用 Top-K 对模型参数进行筛选，对选择的候选索引采用二进制比特压缩。TKP 只保留变化幅度最大的前 k 个梯度，并将选择的分量的位置保存在 L。现有基于双云服务器的方案采用秘密分享实现安全合并算法，而在同态上使用安全合并算法的计算开销是非常巨大的，且单云服务器无法支持，因此本文采用计数加量化的方案，基于统计特征实现单云服务器下的密态安全合并算法，通过计数确定下一轮需要更新的梯度，大大降低了以往基于双云服务器安全合并协议的时间开销。

总体流程是通过 TKP 获得 Top-K 梯度选择的索引数组 L，裁剪量化拼接成大整数加密上传至云服务器，安全聚合后通过每 l bit 对应的值来获得选择该梯度的边缘节点数量。

根据 Top-K 梯度选择的候选索引数组 L，将选择的候选索引对应位置设置为 1，其他位置设置为 0。同时采用 l bit 量化编码，将 1 bit 扩展成 l bit，并联合压缩候选索引集合，通过候选索引量化协议将候选索引数组压缩成大整数，其中 l 表示 EN 个数的二进制位数。联合压缩候选索引数组表示为

$$\text{CIQP}(w_i) = \begin{cases} I_j = 2^l I_j + 1, & i \in L \\ I_j = 2^l I_j, & i \notin L \end{cases} \tag{6}$$

其中，I_j 是第 j 个长整数，用来记录边缘节点的梯度选择，l 是 EN 数量的二进制位数，对梯度 w_i 的选择映射到 I_j 的对应位置。

云服务器最后要对所有边缘节点的选择进行聚合，即使所有的 EN 均选择了第 i 个梯度，其对应位置累加和也不会超过 EN 的个数。因此可用 1 bit 来表示是否选择，$l-1$ bit 扩展用于防止溢出。为了进一步降低通信开销，需要批量拼接候选索引选择，因此提出候选索引量化协议（CIQP）量化压缩候选索引数组。

如图 3 所示，假设模型梯度数量 m=5，边缘节点数量为 3，则 l 为 2 即可满足所有边缘节点的选择之和，前 3 行分别代表每个边缘节点的所有梯度选择，每 2 个框代表一个梯度选择（低位为是否选择该梯度，高位为比特扩展防止溢出）。

根据式(6)，每个边缘节点对要选择的梯度索引位置进行标记，如果当前梯度包含在 L 中，则将该梯度对应的选择位标记为 1，否则标记为 0。云服务器聚合后的计算结果表示当前选择该梯度的边缘节点数量，如果仅使用 1 bit 则不包含比特扩展，在聚合后会溢出导致计算结果错误。不失一般性，各边缘节点根据 Top-K 将梯度变化程度最大的梯度索引位置设置为 1，否则为 0，并将扩展比特 $l-1$ 设置为 0。

每个边缘节点将自己对各梯度的选择拼接后加密上传至云服务器。通过在 CP 上进行同态加法，获得密文结果的明文值，如图 3 中计算结果所示，最后各 EN 获得密文结果解密反量化获得需要上传的梯度。具体算法过程如算法 2 所示。

算法 2　候选索引量化协议（CIQP）

输入　执行 TKP 算法获得候选索引数组 L，一个大整数可存梯度的个数 q，EN 个数对应的二进制位数 l，模型包含的梯度个数 m

输出　量化后的候选索引集合 $\boldsymbol{I}$

1) $n = \dfrac{m}{q}$；量化后索引集合包含大整数的个数

2) 根据 n 初始化索引集合 $\boldsymbol{I}$，$\boldsymbol{I} = [0,0,\cdots,0]_n$；

3) for i=1:1:q

4)　　for j=1:1:n

5)　　$k = i + qj$，$k \in [1,m]$；

6)　　if $k \in L$

7)　　$I_j = 2^l I_j + 1$；

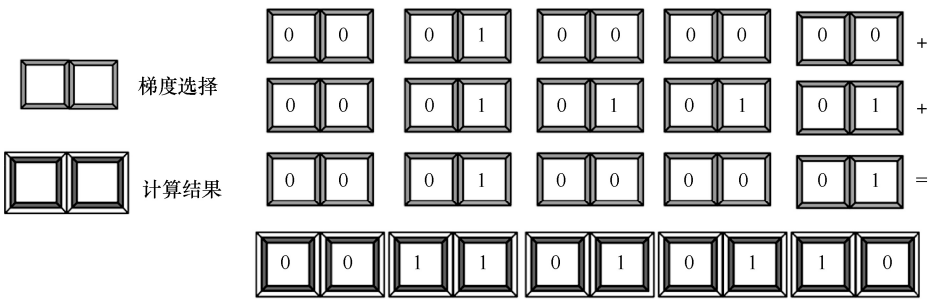


图 3　EN=3 的候选索引合并

8) else

9) $I_j = 2^l I_j$；

10) end if

11) end for

12) end for

步骤 1 EN 通过 TKP 获取模型的候选索引数组 L。

步骤 2 EN 计算量化后的大整数个数。

步骤 3 根据 Top-K 梯度选择的候选索引数组 L，判断当前位置是否为选中的梯度索引，如果为梯度索引，则将当前对应大整数的相对位置即 $2^{l(j-1)}$ 置 1；否则置 0，表示当前参数位置不选择。

每个 EN 上传从 CIQP 获得的候选索引集合 $\boldsymbol{I}$，并通过 paillier 加密候选索引集合，加密后的候选索引集合表示为 $\|\boldsymbol{I}\| = (\llbracket I_1 \rrbracket, \llbracket I_2 \rrbracket, \cdots, \llbracket I_n \rrbracket)$，其中集合长度为 n，$\|\cdot\|$ 表示经过同态加密的数组。CP 接收来自各 EN 上传的密态候选索引集合并进行安全聚合，从而更新全局候选索引集合的值。之后将全局候选索引集合传递给每个 EN，EN 通过 sk 解密以及反量化操作得到候选索引并集。具体算法过程如算法 3 所示。

算法 3 安全候选索引合并协议（SLUP）

输入 加密后的候选索引集合 $\|\boldsymbol{I}\|$，EN 的数量 m，$\|\boldsymbol{I}\|$ 的长度 n，第 i 个 EN 的第 j 个加密的标签值 $\llbracket I_j^i \rrbracket$

输出 聚合后候选索引向量 $\|\boldsymbol{H}\|$

1) EN 执行 CIQP 并量化后获得候选索引集合 $\boldsymbol{I}$；

2) CP 接收来自所有 EN 上传的加密后的候选索引集合 $\|\boldsymbol{I}\|$；

3) 初始化向量 $\boldsymbol{H}$，$\|\boldsymbol{H}\| = (\llbracket 0 \rrbracket, \llbracket 0 \rrbracket, \cdots, \llbracket 0 \rrbracket)_n$；

4) for i=1:1:m

5) for j=1:1:n

6) $\llbracket H_j \rrbracket = \llbracket I_j^i \rrbracket \llbracket H_j \rrbracket = \llbracket I_j^i + H_j \rrbracket$;

7) end for

8) end for

9) CP 发送 $\|\boldsymbol{H}\|$ 到各 EN。

步骤 1 EN 执行 CIQP 得到候选索引集合。

步骤 2 CP 接收来自各 EN 的候选索引集合，其中每个标签存储了多个位置的信息。

步骤 3 CP 通过同态加密对接收的各 EN 的 $\|I^i\|$ 进行安全聚合得到 $\|\boldsymbol{H}\|$，若 $\llbracket H_j \rrbracket$ 对应明文值超过 1，则表示当前位置反量化后对应的梯度需要进行上传。

4.4 梯度批量裁剪量化算法

由于神经网络拥有的模型梯度数量约上万个甚至百万个，在联邦学习场景中，边缘节点和云平台的频繁交互会加剧通信负载，高效的通信优化方法将有效解决通信负载问题。现有的研究采用梯度压缩技术来压缩需要传递的数据或加速仅需要乘法的预测[25]以减少分布式/联邦学习训练过程中的网络流量[27,33-34]。然而，这些方法不是针对梯度聚合而设计的，无法有效地对压缩的梯度进行聚合且存在多个边缘节点导致梯度边界非对称、不一致而无法量化的问题。因此需要对梯度进行对称量化，提前放缩裁剪到对称边界上。

现有的裁剪方案主要包括基于剖面和基于分析建模的方法。剖面裁剪的方法根据样本数据集获得一个样本梯度分布，使用标准的测量值评估阈值，如收敛率[26]。但该方法不符合实际需求，主要有以下 3 个原因。首先，在联邦学习中找到一个代表的数据集是困难的，实际应用中边缘节点的数据集通常具有非独立同分布的特性；其次，梯度范围一般随着迭代轮次的增加而缓慢缩小[35]，因此需要不断裁剪和校准；最后，分析结果特定于训练模型和数据集，一旦模型或者数据集被更改，就需要重新对阈值进行评估。基于以上考虑，ESFL 采用分析建模裁剪 dACIQ[13]方式来获得梯度裁剪的阈值，由于模型梯度是带符号的浮点数，当量化位宽为 16 bit 时近乎无损[36]，因此本文主要采用 16 bit 的方案。

然而，Zhang 等[13]的量化方案仅适合 2 个边缘节点的场景，该方案在负数梯度聚合处理时边缘节点数量增多，导致符号位溢出而无法正确反量化，造成训练结果异常。在此基础上，本文提出了梯度无符号量化协议（GUQP），并在多个 EN 下保证 CP 聚合梯度结果的准确性。为确保模型的准确性，处理方式需要满足 2 个条件：1)安全聚合不存在符号位溢出；2)不改变数据区间阈值的绝对值。因此先保留符号位，将梯度的绝对值量化到 r bit，即将 $[-\alpha, 0]$ 和 $[0, \alpha]$ 统一映射到 $[-(2^r-1), 0]$ 和 $[0, 2^r-1]$，再根据梯度的符号位进行无符号处理，将 $[-(2^r-1), 0]$ 和 $[0, 2^r-1]$ 重新映射到 $[2^r, 2^{r+1}-1]$ 和 $[0, 2^r-1]$ 上，具体如式(7)所示。

$$q_k = \frac{|w_k|}{\alpha}(2^r - 1) + \mathrm{ReLU}(-\mathrm{sgn}(w_k))2^{r+1} \tag{7}$$

其中，w_k 是第 k 个梯度；ReLU 是分段线性函数，当输入大于 0 时保留其值，否则为 0；sgn 是获取符号的函数。

之后与梯度索引时的处理类似，根据 EN 的数量 l，预留扩展位防止计算出现溢出问题，并拼接成大整数，降低计算和通信开销。具体协议如算法 4 所示。

算法 4　梯度无符号量化协议（GUQP）

输入　SLUP 获得密态候选索引集合并解密反量化得到的候选索引向量 $\boldsymbol{H}$ 的长度 m，量化后的每个梯度值的比特长度 u，EN 个数的二进制位数 l，一个大整数可存梯度的个数 q

输出　压缩后的梯度向量 $\boldsymbol{p}$；

1) $u=r+1+l$;

2) 计算量化后的大整数个数，$n=\dfrac{m}{q}$；

3) 根据 n 初始化向量 $\boldsymbol{p}$，$\boldsymbol{p}=[0,0,\cdots,0]_n$；

4) for i=1:1:q

5)　for j=1:1:n

6)　$k=i+qj$；

7)　根据式(7)获得量化后的值 q_k；

8)　$p_j=2^u p_j+q_k$；

9)　end for

10) end for

具体的裁剪量化算法（CQP）如算法 5 所示。

算法 5　裁剪量化协议（CQP）

输入　定义当前梯度 $\boldsymbol{G}$，量化比特带宽 r bit，边缘节点数量的二进制位数 l

输出　压缩后的梯度向量 $\boldsymbol{p}$

1) 循环

2) 每个 EN 计算出当前每层梯度 $\boldsymbol{G}$ 的最大值 max、最小值 min 和数量 size；

3) 每个 EN 发送 max、min 和 size 到 CP;

4) until CP 接收到所有 EN 发送过来的数据

5) CP 通过 dACIQ 计算出 α，并发送给每个 EN;

6) 根据 GUQP 获得量化后的候选梯度向量 $\boldsymbol{p}$。

CQP 协议采用 Zhang 等[13]提出的 dACIQ 来裁剪梯度，各 EN 通过计算出每层梯度的极值及式(4)和式(5)推导出 σ，CP 再通过式(3)来确定裁剪阈值 α。最后，根据本文提出的 GUQP 批量量化梯度，得到候选梯度向量 $\boldsymbol{p}$。

4.5　轻量隐私保护联邦学习框架

为了构建高效率和高准确率的隐私保护的联邦学习框架，本文通过上述 TKP、梯度裁剪量化等协议优化了联邦学习中的同态加密过程。

4.5.1　联邦学习局部模型构建

为了降低隐私保护下联邦学习的通信和计算开销，采用 TKP 来优化通信，同时，通过 CIQP 量化候选梯度索引集合，在 CP 上采用 SLUP 确定各 EN 需要上传的候选梯度，每个 EN 再执行 CQP 对模型梯度进行裁剪量化，并采用 GUQP 拼接成大整数进行加密。为了更进一步优化通信和 CP 聚合开销，提出构建局部模型协议（BLMP），如算法 6 所示。

算法 6　构建局部模型协议（BLMP）

输入　量化比特带宽 r，每 k 轮上传局部模型，公钥 pk，私钥 sk，本地迭代轮次为 E，训练的批处理大小 T，第 i 个 EN（EN^i）

输出　最优局部模型梯度 $\boldsymbol{G}^i$

1) for e=1:1:E

2)　for t=1:1:T

3)　EN^i 计算并更新梯度 $\boldsymbol{G}^i$;

4)　end for

5) if e mod k == 0

6) $\Delta\boldsymbol{G}^i=\boldsymbol{G}^i-\boldsymbol{G}^{\mathrm{glob}}$；

7) 执行 TKP 获得候选索引数组 L；

8) 执行 CIQP 和 SLUP 获得候选索引向量 $\boldsymbol{H}$；

9) 根据候选索引向量 $\boldsymbol{H}$，对选择的梯度执行 CQP 获得裁剪量化的候选梯度 $\Delta\boldsymbol{G}^i$，并加密成 $\left\|\Delta\boldsymbol{G}^i\right\|$；

10) 发送 $\left\|\Delta\boldsymbol{G}^i\right\|$ 到 CP 进行聚合；

11) 接收来自 CP 的 $\left\|\Delta\boldsymbol{G}^{\mathrm{glob}}\right\|$；

12) 解密 $\left\|\Delta\boldsymbol{G}^{\mathrm{glob}}\right\|$ 并反量化；

13) 更新 EN^i 局部模型 $\boldsymbol{G}^i$；

14) $\boldsymbol{G}^{\mathrm{glob}}$=$\boldsymbol{G}^i$;

15) end if

16) end for

步骤 1　每个 EN 都会接收到来自 KGC 生成的公钥 pk 和私钥 sk，对需要发送到 CP 上的敏感信息使用 pk 进行加密，对收到来自 CP 的密文使用 sk 解密得到明文。

步骤 2　每次迭代根据批处理训练结果更新局部模型 $\boldsymbol{G}^i$。

步骤 3　当轮次 e 正好是 k 的倍数时，计算出 $\Delta\boldsymbol{G}^j$，执行步骤 4～步骤 8。

步骤 4 TKP 选择出 k 个梯度值对应的候选索引数组 L，在每个 EN 上执行 CIQP 对候选索引数组 L 进行量化拼接成大整数，并通过 SLUP 获得所有 EN 的候选梯度索引并集。

步骤 5 在 EN 上，根据 SLUP 获得的候选索引集合，将 $\Delta \boldsymbol{G}^i$ 和量化批处理大小发送到 CP，CP 与 EN^i 联合执行 CQP 计算出裁剪阈值 α。对于每个 EN，使用阈值 α 来对梯度进行裁剪和量化。

步骤 6 对裁剪量化后的梯度通过 pk 进行加密，并发送给 CP。

步骤 7 各 EN 接收来自 CP 的全局模型，并通过 sk 进行解密和反量化操作。

步骤 8 根据反量化后的梯度值，更新各 EN 的局部模型。

步骤 9 重复执行步骤 2～步骤 8，直到迭代结束。

4.5.2 联邦学习全局模型更新

全局模型更新协议（GMUP）如算法 7 所示，使 CP 和 EN 之间可以安全地构建全局模型，并有效地更新每个 EN 的局部模型。CP 从 N 个 EN 接收到局部模型梯度，并使用同态加密技术进行模型聚合，得到全局模型。CP 将全局模型发送到各 EN，通过解密和反量化得到模型梯度，并更新局部模型。由于局部模型梯度是经过多个梯度量化后拼接的大整数，对每个梯度都预留了扩展位，CP 仅需进行同态加法就可以实现对应梯度的聚合，且能保证其正确性。

算法 7 全局模型更新协议（GMUP）

输入 EN 的数量 n，模型包含梯度的个数 m，EN^i 的模型 $\Delta \boldsymbol{G}^i$，EN^i 的模型第 j 个梯度 $\Delta \boldsymbol{G}_j^i$

输出 全局模型梯度 $\left\|\Delta \boldsymbol{G}^{\mathrm{glob}}\right\|$

1) for $j=1:1:n$

2) CP 接收来自所有 EN^i 上传的加密模型参数增量 $\left\|\Delta \boldsymbol{G}^i\right\|$；

3) end for

4) 初始化全局模型梯度 $\left\|\Delta \boldsymbol{G}^{\mathrm{glob}}\right\|$，$\left\|\Delta \boldsymbol{G}^{\mathrm{glob}}\right\|=\left[\left[0\right],\left[0\right],\cdots,\left[0\right]\right]_m$；

5) for $j=1:1:m$

6) $\left[\Delta \boldsymbol{G}_j^{\mathrm{glob}}\right]=\left[\Delta \boldsymbol{G}_j^{\mathrm{glob}}\right]\left[\Delta \boldsymbol{G}_j^i\right]=\left[\Delta \boldsymbol{G}_j^{\mathrm{glob}}+\Delta \boldsymbol{G}_j^i\right]$；

7) end for

8) 发送 $\left\|\Delta \boldsymbol{G}^{\mathrm{glob}}\right\|$ 到所有 EN。

步骤 1 CP 接收来自各 EN 发来的模型梯度 $\left\|\Delta \boldsymbol{G}^i\right\|$。

步骤 2 对每个 EN，通过同态加密的加法聚合 $\left\|\Delta \boldsymbol{G}^i\right\|$ 得到 $\left\|\Delta \boldsymbol{G}^{\mathrm{glob}}\right\|$。

步骤 3 发送聚合后的全局模型 $\Delta \boldsymbol{G}^{\mathrm{glob}}$ 到各 EN。

5 安全证明

本节分析所支持的安全性协议和所提出的 ESFL 框架，特别地，描述了该系统基于各种潜在对手的安全性。在诚实且好奇的模型中，使用以下定义和定理证明协议是安全的。

定义 1 假设对手 $\mathcal{A}$ 在现实世界与模拟器 Sim 交互以完成理想世界中的计算过程。加密数据在 CP 上的 $\left\|X\right\|$ 是安全输入，如果 ESFL 模型是安全的，则可以表示为 $\left\{\mathrm{IDEAL}_{\Pi,\mathrm{Sim}}\left(\left\|X\right\|\right)\right\},\equiv_c\left\{\mathrm{REAL}_{\Pi,\mathcal{A}}\left(\left\|X\right\|\right)\right\}$。其中，$\left\|\cdot\right\|$ 表示同态加密后的数组，Π 表示对应的协议，$\equiv_c$ 表示在计算上是不可区分的。

引理 1 协议 TKP、CIQP 和 GUQP 在 EN 本地操作，是安全的。

引理 2 遵循 paillier 同态加密的所有基础操作均视为安全的。

引理 3 如果一个协议的所有子协议是完美的模拟，即不可区分，则该协议是不可区分的。

定理 1 在半诚实模型（非碰撞）下，即使存在对手 $\mathcal{A}_{\mathrm{CP}}$ 的威胁，安全候选索引合并协议 (SLUP) 仍然可保证安全。

证明 假设 $\mathcal{A}_{\mathrm{CP}}$ 将损坏服务器 CP，本文将构造模拟器 $\mathrm{Sim}_{\mathrm{CP}}$，在理想世界中执行，其中 $\mathrm{Sim}_{\mathrm{CP}}$ 的构造如下。

对于 $\mathrm{Sim}_{\mathrm{CP}}$，协议执行中的视图将是 $\mathrm{View}_{\mathrm{CP}}=(\left\|\boldsymbol{I}\right\|,\left\|\boldsymbol{H}\right\|)$。根据引理 2，在 CP 上，计算 $\left[\!\left[H_j\right]\!\right]=\left[\!\left[I_j^i\right]\!\right]\left[\!\left[H_j\right]\!\right]=\left[\!\left[I_j^i+H_j\right]\!\right]$，根据各 EN 发送的候选索引数组，在 CP 上进行安全聚合，由于同态加密的加法操作是安全的，并无任何隐私信息泄露，同时将最后的计算结果发送给每个 EN，因此 SLUP 在实际执行和理想执行中是无法区分的。综合以上分析，模拟器 $\mathrm{Sim}_{\mathrm{CP}}$ 生成一种在计算上与实际无法区分的视图，因此 SLUP 在理想和现实中是无法区分的理想执行。

定理 2 在半诚实模型（非碰撞）下，即使存在对手 $\mathcal{A}_{\mathrm{CP}}$ 的威胁，裁剪量化协议（CQP）仍然可保证安全。

证明 假设 $\mathcal{A}_{\mathrm{CP}}$ 将损坏服务器 CP，本文将构造模拟器 $\mathrm{Sim}_{\mathrm{CP}}$，在理想世界中执行，其中 $\mathrm{Sim}_{\mathrm{CP}}$ 的

构造如下。

对于 $\mathrm{Sim_{CP}}$，Zhang 等[13]已证明 dACIQ 计算过程中的安全性，而对于 GUQP，由引理 1 可知，其计算过程仅在 EN 本地上，因此 CP 无法获得任何信息，综合以上分析，模拟器 $\mathrm{Sim_{CP}}$ 将生成一种在计算上与实际无法区分的视图，CQP 在理想和现实中是无法区分的理想执行。

定理 3　在半诚实模型（非碰撞）下，即使存在对手 $\mathcal{A}_{\mathrm{CP}}$ 的威胁，构建局部模型协议（BLMP）仍然可保证安全。

证明　假设 $\mathcal{A}_{\mathrm{CP}}$ 将损坏服务器 CP，本文将构造模拟器 $\mathrm{Sim_{CP}}$，在理想世界中执行，其中 $\mathrm{Sim_{CP}}$ 的构造如下。

对于 $\mathrm{Sim_{CP}}$，协议执行中的视图为 $\mathrm{View_{CP}}=\left\|\Delta \boldsymbol{G}^{i}\right\|$。仅在 EN 本地上进行计算，其操作是满足安全的，只需要保证 CP 与 EN 之间的交互满足安全性。由于每轮执行的操作是重复的，因此可以根据一轮 CP 和 EN 之间操作的安全性，将整个迭代视为安全的。当 e 正好被本地迭代更新轮次 k 整除时，将执行 CP 与 EN 之间的交互，根据引理 1、引理 3 和定理 1～定理 2 的证明可知，TKP、CIQP、CQP 和 SLUP 均是安全的。EN 是以密文形式将局部模型梯度发送到 CP 的，CP 无法获得任何信息。综合以上分析，模拟器 $\mathrm{Sim_{CP}}$ 将生成一种在计算上与实际无法区分的视图，BLMP 在理想和现实中是无法区分的理想执行。

定理 4　在半诚实模型（非碰撞）下，即使存在对手 $\mathcal{A}_{\mathrm{CP}}$ 的威胁，全局模型更新协议（GMUP）仍然可保证安全。

证明　假设 $\mathcal{A}_{\mathrm{CP}}$ 将损坏服务器 CP，本文将构造模拟器 $\mathrm{Sim_{CP}}$，在理想情况中执行，其中 $\mathrm{Sim_{CP}}$ 的构造如下。

对于 $\mathrm{Sim_{CP}}$，协议执行中的视图为 $\mathrm{View_{CP}}=\left\|\Delta \boldsymbol{G}^{l}\right\|$。根据引理 2，在 CP 上计算 $\left[\!\left[\Delta \boldsymbol{G}_{j}^{\mathrm{glob}}\right]\!\right]=\left[\!\left[\Delta \boldsymbol{G}_{j}^{\mathrm{glob}}\right]\!\right]\left[\!\left[\Delta \boldsymbol{G}_{j}^{i}\right]\!\right]$，根据各 EN 发送过来的局部模型梯度进行安全聚合，因此 CMUP 在实际执行和理想执行中是无法区分的。综合以上分析，模拟器 $\mathrm{Sim_{CP}}$ 将生成一种在计算上与实际无法区分的视图，因此，CMUP 在理想和现实中是无法区分的理想执行。

综合以上分析，推测分布 $\mathrm{REAL}_{\Pi,\mathcal{A}_{\mathrm{CP}}}$ 和 $\mathrm{IDEAL}_{\Pi,\mathrm{Sim_{CP}}}$ 是无法区分的，根据定理 1～定理 4 的证明，模拟器 $\mathrm{Sim_{CP}}$ 很容易生成一种在计算上与实际无法区分的视图，则有 $\mathrm{IDEAL}_{\Pi,\mathrm{Sim_{CP}}} \equiv_c \mathrm{REAL}_{\Pi,\mathcal{A}_{\mathrm{CP}}}$。因此，ESFL 在真实和理想的计算过程中是无法区分的。

6　实验性能评估

本节实验在不同参数设置的神经网络模型下详细讨论 ESFL 性能，并通过训练的真实神经网络模型来评估 ESFL 的性能。

6.1　实验设置

本文实验的服务器环境为 Intel(R) Xeon(R) Gold 64 内核，2.3 GHz，128 GB 内存，Ubuntu 16.04。实验使用了 MNIST 数据集[37]和 CIFAR10 数据集[38]。

本文考虑 ESFL 的设置基于不同的边缘节点数量（5、10、20 和 50）和不同的 Top-K（1%、5%、10%和 20%）策略，通过准确率、通信开销、计算开销和压缩率来衡量方案的有效性。

为了证明提出的 ESFL 的有效性，分别在不同神经网络模型上进行一系列实验。MNIST 数据集使用的是 CNN，其结构主要包括全连接层、池化层、ReLU 函数和卷积层。CIFAR10 数据集使用的是 LeNet5 网络，主要包括全连接层、池化层、ReLU 函数层和卷积层。本文采用的同态加密方案是 paillier，并使用 joblib 并行加速同态计算。

6.2　基线

本文比较了 Zhang 等[13]的 BatchCrypt 方案、明文模型，以及不同量化位宽在 ESFL 上的效率和准确率，对不同的 Top-K 进行对比，并分析了量化方案对训练准确率的影响。

6.3　数据集

基于 2 个图像识别任务来评估 ESFL 的性能：MNIST[37]的数字识别任务和 CIFAR10[38]的图像分类任务。MNIST 数据集由 10 个类组成，包括 60 000 个图像训练样本和 10 000 个图像测试样本，且每个图像样本都是 28 像素×28 像素；CIFAR10 数据集由 10 个类组成，包括 60 000 个图像训练样本和 10 000 个图像测试样本，且每个图像样本都是 32 像素×32 像素彩色图像。

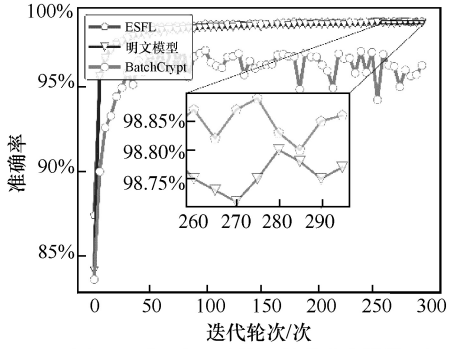
6.4　实验性能

6.4.1　模型准确率

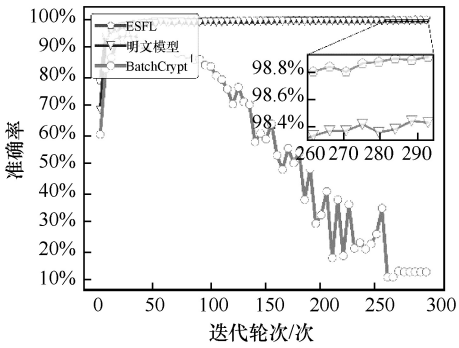
为了衡量模型的性能，本文通过在测试集上测试模型的准确率变化，表明 ESFL 对准确率影响是可接受的。

首先对不同方案进行性能对比。实验量化位宽 r 分别设置为 8 bit、16 bit 和 32 bit，边缘节点数量分别为 5、10、20 和 50，Top-K 梯度选择比率为 10%，批处理大小为 128，同态加密安全参数为 2 048，并将结果与明文模型（纯分布式学习，采用相同边缘节点数量和 Top-K 梯度选择比率，不涉及加密）和 BatchCrypt 方案进行比较。

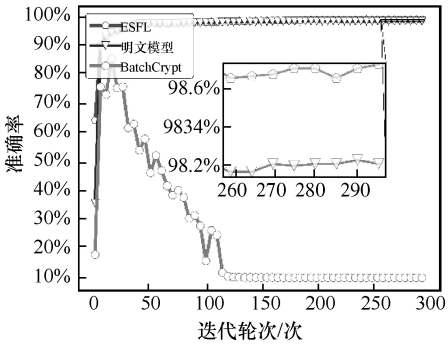
实验结果如图 4 所示。从图 4(a)～图 4(d)可知，



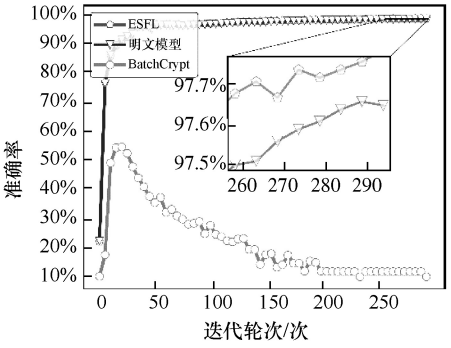
(a) MNIST（Top-10%，边缘节点数量=5）



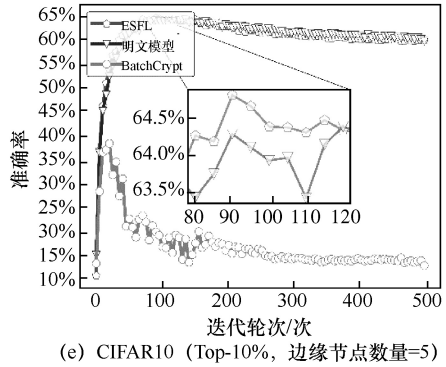
(b) MNIST（Top-10%，边缘节点数量=10）



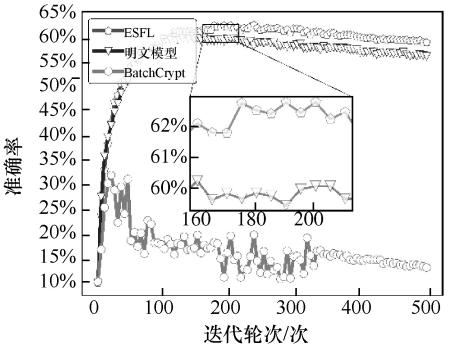
(c) MNIST（Top-10%，边缘节点数量=20）



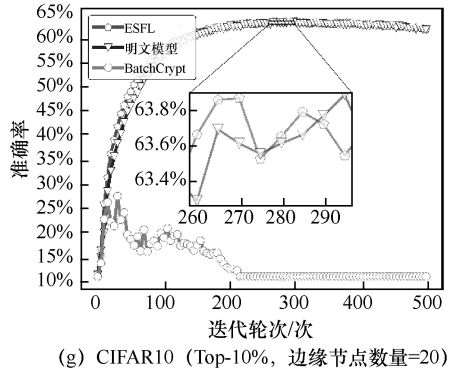
(d) MNIST（Top-10%，边缘节点数量=50）



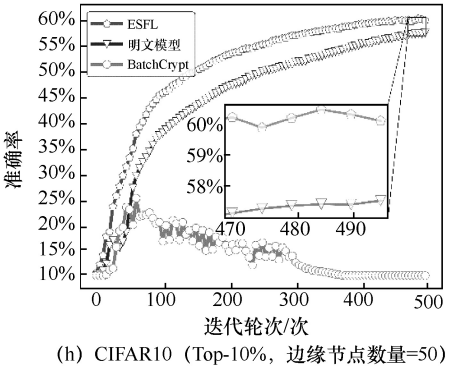
(e) CIFAR10（Top-10%，边缘节点数量=5）



(f) CIFAR10（Top-10%，边缘节点数量=10）



(g) CIFAR10（Top-10%，边缘节点数量=20）



(h) CIFAR10（Top-10%，边缘节点数量=50）

图 4　不同方案的性能对比

在 MNIST 数据集上，BatchCrypt 方案无法保证准确率，随着边缘节点数量增多，其准确率逐渐降低，最低峰值为 72.31%。从图 4(e)～图 4(h)可知，在 CIFAR10 数据集上，BatchCrypt 方案最高峰值不超过 40.36%，远低于明文模型上的准确率。为了进一步验证 BatchCrypt 方案的准确率，采用 Top-5%的设置，具体结果如表 1 所示。从表 1 可知，当 MNIST 数据集上边缘节点数量较少时，可保证相近的准确率。随着边缘节点数量增多，其峰值仅为 73.68%，与明文模型相差 24.05%，而在 CIFAR10 数据集上与明文模型准确率的差距甚至达到了 32.76%，由于 BatchCrypt 方案无法保证准确率，并不适合多边缘节点场景。在 16 bit 的方案中，即使边缘节点数量上升，ESFL 依旧保持着与明文模型相近的准确率，且 ESFL 在 MNIST 和 CIFAR10 数据集上准确率曲线整体吻合，差异较小。

为了验证不同边缘节点数量对准确率的影响，将边缘节点数量分别设置为 5、10、20 和 50，Top-K 梯度选择比率为 1%，批处理大小为 128，同态加密安全参数为 2 048，将明文模型结果作为基线进行比较。从图 5 可知，各神经网络在不同数据集上的准确率随着边缘节点数量的上升而下降，当边缘节点数量为 50 时，8 bit 的量化位宽变化程度最大，准确率甚至低于明文模型 10%；16 bit 和 32 bit 的量化位宽下，模型的准确率不会因为边缘节点数量的增多而降低，其甚至比明文模型的准确率更高。

表 1　不同方案 Top-5%准确率峰值

数据集	边缘节点数量/个	方案准确率		
		ESFL(16 bit)	明文模型	BatchCrypt (16 bit)
MNIST	5	98.90%	98.79%	97.61%
	10	98.01%	98.44%	94.16%
	20	97.92%	98.45%	90.27%
	50	97.33%	97.73%	73.68%
CIFAR10	5	64.43%	64.80%	47.44%
	10	62.93%	60.27%	37.87%
	20	63.86%	58.77%	29.11%
	50	61.31%	59.82%	27.06%

为了验证不同量化位宽 r 对准确率的影响，将 r 分别设置为 8 bit、16 bit 和 32 bit，Top-K 梯度选择比率为 1%，批处理大小为 128，同态加密安全参数为 2 048，将明文模型结果作为基线进行比较。从图 5(d)和图 5(h)中可以看出，在 r =8 bit 的量化位宽方案下，其训练的准确率与明文模型相差较大，而图 5(a)～图 5(h)中 16 bit 的量化位宽方案在不同边缘节点数量下均与明文模型相差不大，且 32 bit 量化位宽方案在 MNIST 和 CIFAR10 数据集上均略微高于明文模型。然而，为了兼顾通信开销以及模型准确率，16 bit 的量化位宽方案相比 32 bit 的量化位宽方案压缩了更多的通信量，同时保证了模型计算的准确率，相比而言，8 bit 的量化位宽方案准确率较低，无法保证模型的稳健性。因此采用 16 bit 的量化位宽方案既可以保证较好的训练结果，又可以压缩联邦学习各边缘节点上传加密模型梯度的开销。

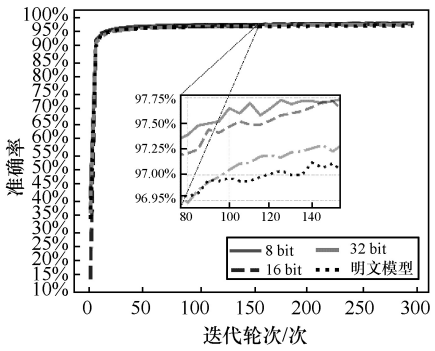
6.4.2　时间和通信开销

时间和通信开销是联邦学习的瓶颈，当边缘节点将数量巨大的梯度全部上传时，服务器需要承受极大的通信和计算负担。在此基础上，ESFL 通过 Top-K 梯度选择提高通信性能，采用量化方案对多边缘节点的 Top-K 候选梯度进行量化来减少交互过程的通信数据量，此量化方案将减少各边缘节点本地的数据加密次数以及服务器安全聚合的时间，其中，

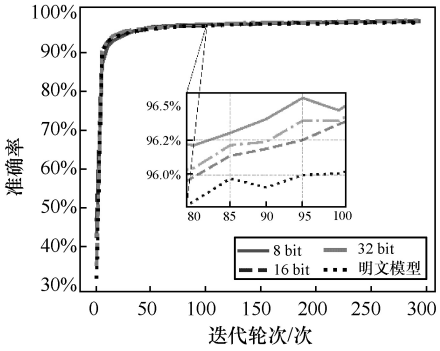
$$\text{压缩率}=\frac{\text{密态模型通信量}-\text{比特位宽通信量}}{\text{密态模型通信量}}。$$

为了验证 ESFL 的通信和时间性能，设置同态加密安全参数为 2 048，批处理大小为 128，边缘节点数量为 5，r 分别为 8 bit、16 bit 和 32 bit，在 Top-1%、Top-5%、Top-10%和 Top-20%上，与董业等[14]提出的高效联邦聚合的密态模型（采用相同的实验设置）进行比较。

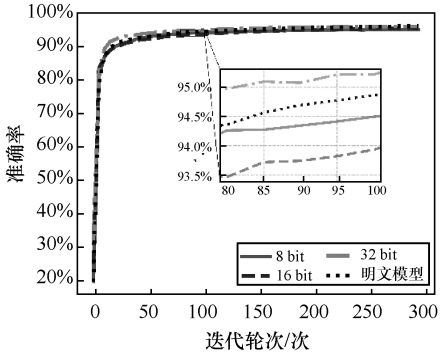
从图 6 可知，在 MNIST 数据集上，8 bit 量化位宽方案压缩率为 84%～87%，16 bit 量化位宽方案压缩率为 74%～81%，32 bit 量化位宽方案压缩率为 70%～76%，而在 CIFAR10 数据集上，8 bit 量化位宽方案压缩率为 90%左右，16 bit 量化位宽方案压缩率为 79%～83%，32 bit 量化位宽方案压缩率为 74%～78%，通信效率提升巨大。通过实验测试，采用量化方案的 CIFAR10 数据集的单次聚合时间为 0.01～0.10 s，MNIST 数据集的单次聚合时间为 0.004～0.070 s，聚合效率同样得到了很大的提升。



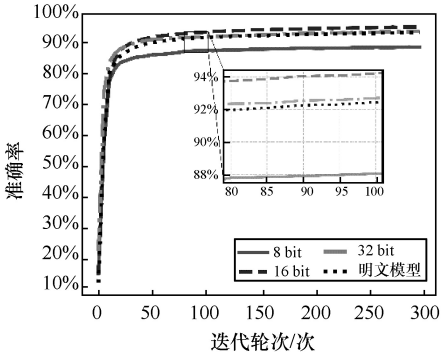
（a）MNIST（Top-1%，边缘节点数量=5）



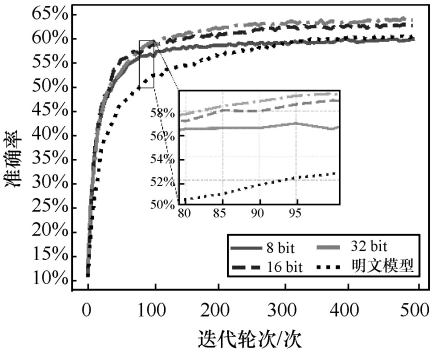
（b）MNIST（Top-1%，边缘节点数量=10）



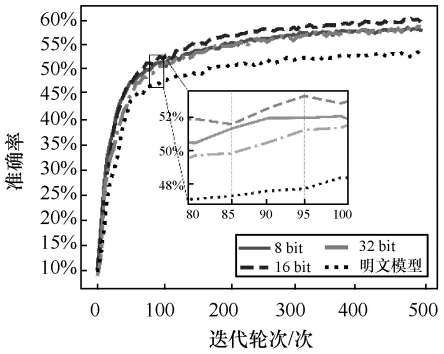
（c）MNIST（Top-1%，边缘节点数量=20）



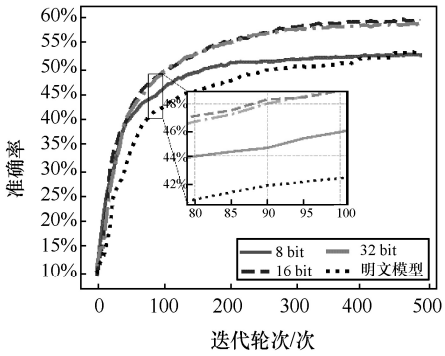
（d）MNIST（Top-1%，边缘节点数量=50）



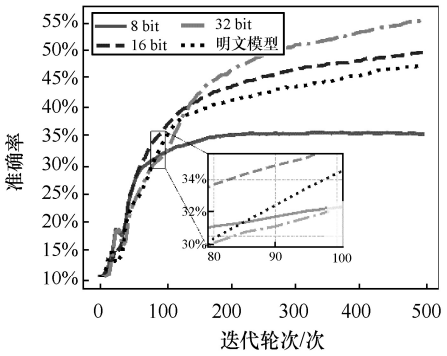
（e）CIFAR10（Top-1%，边缘节点数量=5）



（f）CIFAR10（Top-1%，边缘节点数量=10）



（g）CIFAR10（Top-1%，边缘节点数量=20）



（h）CIFAR10（Top-1%，边缘节点数量=50）

图 5　ESFL 模型性能分析

7　结束语

本文提出了一个基于同态加密的高效联安全邦聚合框架 ESFL，旨在隐私保护场景下，解决 CP 与 EN 之间巨大的通信和计算开销问题。基于 Top-K 优化通信的方法，设计了候选梯度索引量化和安全候选索引

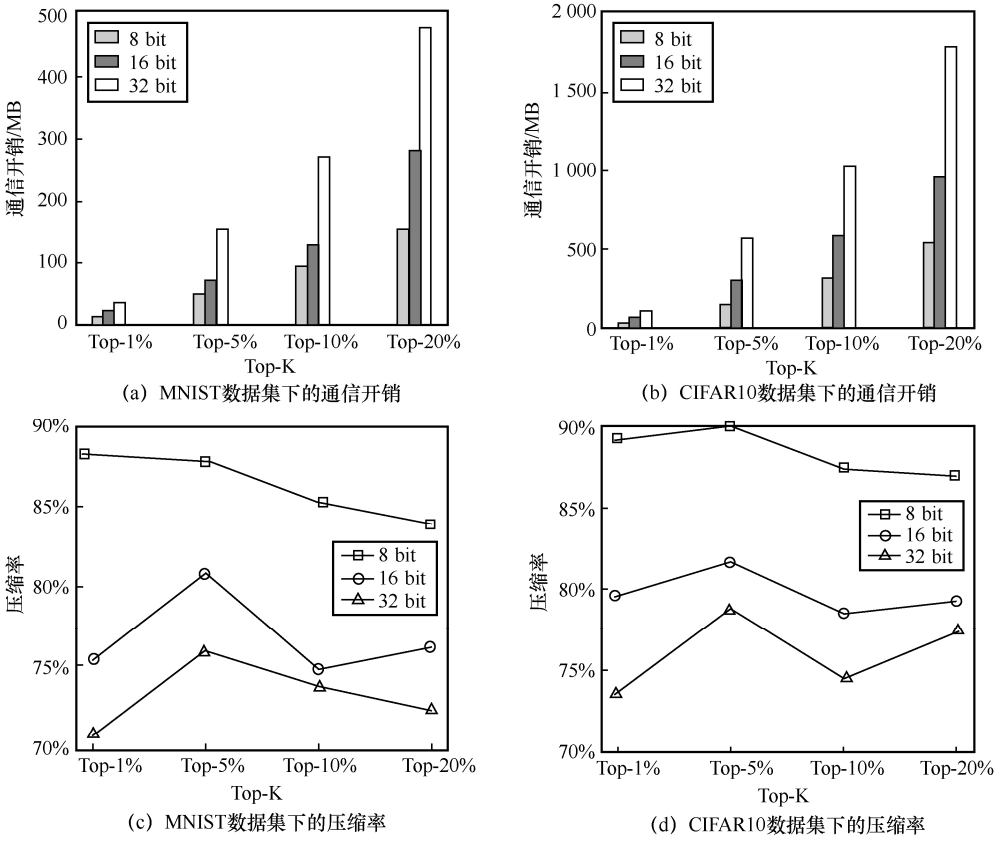


图 6　不同量化比特位宽通信开销与压缩率

合并协议，安全高效地实现了 CP 与多个 EN 之间的交互。考虑多个 EN 情况下的 CP 聚合效率问题和模型准确性问题，本文提出了新的梯度无符号量化方案，解决了现有方案无法在多边缘节点下保证计算正确性的问题。实验结果表明，本文提出的 ESFL 具有较高的性能，在降低通信开销的同时保证了模型准确率。

参考文献：

[1] TEAM I G P. EU General data protection regulation (GDPR) - an implementation and compliance guide, fourth edition[M]. Cambridgeshire: IT Governance Publishing, 2020.

[2] EVANS D, KOLESNIKOV V, ROSULEK M. A pragmatic introduction to secure multi-party computation[J]. Foundations and Trends in Privacy and Security, 2018, 2(2/3): 70-246.

[3] DWORK C, ROTH A. The algorithmic foundations of differential privacy[J]. Foundations and Trends in Theoretical Computer Science, 2013, 9(3/4): 211-407.

[4] ACAR A, AKSU H, ULUAGAC A S, et al. A survey on homomorphic encryption schemes[J]. ACM Computing Surveys, 2019, 51(4): 1-35.

[5] LIU X M. Hybrid privacy-preserving clinical decision support system in fog-cloud computing[J]. Future Generation Computer Systems, 2018, 78: 825-837.

[6] CHEN Z K, ZHENG Z W, LIU X M, et al. Privacy-preserving computation tookit on floating-point numbers[C]//Mobile Multimedia Communications. Berlin: Springer, 2021: 462-476.

[7] HU S S, LI M H, WANG Q, et al. Outsourced biometric identification with privacy[J]. IEEE Transactions on Information Forensics and Security, 2018, 13(10): 2448-2463.

[8] MA Z R. Privacy-preserving and high-accurate outsourced disease predictor on random forest[J]. Information Sciences, 2019, 496: 225-241.

[9] LIU X M, CHOO K K R, DENG R H, et al. Efficient and privacy-preserving outsourced calculation of rational numbers[J]. IEEE Transactions on Dependable and Secure Computing, 2018, 15(1): 27-39.

[10] LI T, SAHU A K, TALWALKAR A, et al. Federated learning: challenges, methods, and future directions[J]. IEEE Signal Processing Magazine, 2020, 37(3): 50-60.

[11] BHOWMICK A, DUCHI J, FREUDIGER J, et al. Protection against reconstruction and its applications in private federated learning[J]. arXiv Preprint, arXiv: 1812.00984, 2018.

[12] MELIS L, SONG C Z, CRISTOFARO E D, et al. Exploiting unintended feature leakage in collaborative learning[C]//Proceedings of 2019 IEEE Symposium on Security and Privacy. Piscataway: IEEE Press, 2019: 691-706.

[13] ZHANG C, LI S, XIA J, et al. BatchCrypt: efficient homomorphic encryption for Cross-Silo federated learning[C]//Proceedings of 2020 USENIX Annual Technical Conference. Berkeley: USENIX Association, 2020: 493-506.

[14] 董业, 侯炜, 陈小军, 等. 基于秘密分享和梯度选择的高效安全联邦学习[J]. 计算机研究与发展, 2020, 57(10): 2241-2250.
DONG Y, HOU W, CHEN X J, et al. Efficient and secure federated learning based on secret sharing and gradients selection[J]. Journal of Computer Research and Development, 2020, 57(10): 2241-2250.

[15] STROM N. Scalable distributed DNN training using commodity GPU cloud computing[C]//Proceedings of Interspeech 2015. Piscataway: IEEE Press, 2015: 1488-1492.

[16] DRYDEN N, MOON T, JACOBS S A, et al. Communication quanti-

zation for data-parallel training of deep neural networks[C]//Proceedings of 2016 2nd Workshop on Machine Learning in HPC Environments (MLHPC). Piscataway: IEEE Press, 2016: 1-8.
[17] AJI A F, HEAFIELD K. Sparse communication for distributed gradient descent[C]//Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Boston: Association for Computational Linguistics, 2017: 440-445.
[18] ALISTARH D, LI J, TOMIOKA R, et al. QSGD: randomized quantization for communication-optimal stochastic gradient descent[J]. arXiv Preprint, arXiv: 1610.02132, 2016.
[19] BONAWITZ K, IVANOV V, KREUTER B, et al. Practical secure aggregation for privacy-preserving machine learning[C]//Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2017: 1175-1191.
[20] SHAMIR A. How to share a secret[J]. Communications of the ACM, 1979, 22(11): 612-613.
[21] NIU C Y, WU F, TANG S J, et al. Secure federated submodel learning [J]. arXiv Preprint, arXiv: 1911.02254, 2019.
[22] BLOOM B H. Space/time trade-offs in hash coding with allowable errors[J]. Communications of the ACM, 1970, 13(7): 422-426.
[23] DONG Y, CHEN X, SHEN L, et al. EaSTFLy: efficient and secure ternary federated learning[J]. Computers & Security, 2020, 94: 101824.
[24] PAILLIER P. Public-key cryptosystems based on composite degree residuosity classes[C]//Advances in Cryptology - EUROCRYPT '99. Berlin: Springer, 1999: 223-238.
[25] BANNER R, NAHSHAN Y, SOUDRY D. Post training 4-bit quantization of convolutional networks for rapid-deployment[C]//Proceedings of the 33rd International Conference on Neural Information Processing Systems. New York: Curran Associates Inc., 2019: 950-7958.
[26] WEN W, XU C, YAN F, et al. TernGrad: ternary gradients to reduce communication in distributed deep learning[C]//Proceedings of the 31st International Conference on Neural Information Processing Systems. New York: Curran Associates Inc., 2017: 1508-1518.
[27] SONG L, ZHAO K, PAN P, et al. Communication efficient SGD via gradient sampling with Bayes prior[C]//Proceedings of 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway: IEEE Press, 2021: 12060-12069.
[28] ANDERSON A G, BERG C P. The high-dimensional geometry of binary neural networks[J]. arXiv Preprint, arXiv: 1705.07199, 2017.
[29] SOUDRY D, HUBARA I, MEIR R. Expectation backpropagation: parameter-free training of multilayer neural networks with continuous or discrete weights[C]//Proceedings of the 27th International Conference on Neural Information Processing Systems. Massachusetts: MIT Press, 2014: 963-971.
[30] BANNER R, HUBARA I, HOFFER E, et al. Scalable methods for 8-bit training of neural networks[C]//Proceedings of the 32nd International Conference on Neural Information Processing Systems. New York: Curran Associates Inc., 2019: 5151-5159.
[31] TANG H L, LIAN X R, ZHANG T, et al. DoubleSqueeze: parallel stochastic gradient descent with double-pass error-compensated compression[C]//International Conference on Machine Learning. New York: PMLR, 2019: 6155-6165.
[32] STICH S U, CORDONNIER J B, JAGGI M. Sparsified SGD with memory[J]. arXiv Preprint, arXiv: 1809.07599, 2018.
[33] KOLOSKOVA A, STICH S U, JAGGI M. Decentralized stochastic optimization and gossip algorithms with compressed communication[C]//International Conference on Machine Learning. New York: PMLR, 2019: 3478-3487.
[34] LIN Y J, HAN S, MAO H Z, et al. Deep gradient compression: reducing the communication bandwidth for distributed training[J]. arXiv Preprint, arXiv: 1712.01887, 2017.
[35] COURBARIAUX M, BENGIO Y, DAVID J. Training deep neural networks with low precision multiplications[J]. arXiv Preprint, arXiv: 1412.7024, 2014.
[36] GUPTA S, AGRAWAL A, GOPALAKRISHNAN K, et al. Deep learning with limited numerical precision[C]//International Conference on Machine Learning. New York: PMLR, 2015: 1737-1746.
[37] LECUN Y, BOTTOU L, BENGIO Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.
[38] KRIZHEVSKY A, NAIR V, HINTON G. The CIFAR-10 dataset[EB]. 2014.

[作者简介]

余晟兴（1995– ），男，福建福州人，北京大学博士生，主要研究方向为机器学习、隐私保护、区块链、可验证计算等。

陈钟（1963– ），男，江苏徐州人，博士，北京大学教授、博士生导师，主要研究方向为网络与信息安全、区块链等。