·热点与综述·　　文章编号：1000-3428(2023)02-0024-07　　文献标志码：A　　中图分类号：TP391

基于茫然传输协议的FATE联邦迁移学习方案

郑云涛[1]，叶家炜[2]

（1.复旦大学 计算机科学技术学院 上海市智能信息处理重点实验室，上海 200433；
2.复旦大学 计算机科学技术学院 教育部网络信息安全审计与监控工程研究中心，上海 200433）

摘　要：利用不同来源的数据参与机器学习模型的训练，能够使得训练出的模型的预测结果更加准确，然而大量的数据收集则会产生隐私方面的相关问题。FATE联邦迁移学习是一种基于同态加密的联邦学习框架，但FATE联邦迁移学习中同态加密计算复杂，收敛速度相对较慢，导致模型训练效率低。提出一种基于茫然传输协议的安全矩阵计算方案。通过实现矩阵加法和乘法及数乘的安全计算，完成参与两方交互下具有数据隐私保护特性机器学习模型的损失函数计算与梯度更新，并以此构造更高效的FATE联邦迁移学习算法方案。在此基础上，通过茫然传输扩展协议和通信批量处理，减少需要调用的茫然传输协议的数量，缩减通信轮数，从而降低茫然传输协议带来的通信消耗。性能分析结果表明，该方案的安全模型满足安全性和隐私保护性，并且具有一定的可扩展性，在局域网环境下，相比基于同态加密的方案，模型收敛的平均时间缩短约25%，并且随着数据样本特征维度的增加，该方案仍能保持稳定的收敛速度。

关键词：联邦迁移学习；安全多方计算；秘密共享；茫然传输协议；同态加密

开放科学（资源服务）标志码（OSID）：

源代码链接：https://github.com/ShiroNekoHouse/MyFate/tree/master/federatedml/ftl

中文引用格式：郑云涛，叶家炜，基于茫然传输协议的FATE联邦迁移学习方案[J].计算机工程，2023，49(2)：24-30.

英文引用格式：ZHENG Y T，YE J W.FATE federated transfer learning scheme based on oblivious transfer protocol[J].Computer Engineering，2023，49(2)：24-30.

FATE Federated Transfer Learning Scheme Based on Oblivious Transfer Protocol

ZHENG Yuntao[1]，YE Jiawei[2]

（1.Shanghai Key Laboratory of Intelligent Information Processing，School of Computer Science，Fudan University，Shanghai 200433，China；
2.Engineering Research Center of Cyber Security Auditing and Monitoring，Ministry of Education，School of Computer Science，Fudan University，Shanghai 200433，China）

【Abstract】Machine learning models can be trained on a large amount of data collected from different sources，thereby improving the prediction accuracy.However，data collection between different organizations causes privacy issues. The Federated Artificial intelligence Technology Enabler（FATE）Federated Transfer Learning（FTL）is a secure machine learning framework based on homomorphic encryption.However，the convergence speed is relatively low owing to the computational efficiency of homomorphic encryption.In this study，a secure multiparty computation scheme for matrix computation based on Oblivious Transfer（OT）protocol is proposed to design a two-party machine-learning scheme to construct a more efficient FATE FTL.In addition，the communication consumption caused by the OT protocol is reduced via OT extension and batch processing.The performance analysis shows that the scheme ensures security，privacy preservation，and scalability in practical applications.On the other hand，the convergence time of the proposed scheme is approximately 25% better than that FATE FTL framework based on a homomorphic encryption scheme in a local-area network environment. As the feature dimension of the data samples increases，the advantage of the convergence speed of this scheme can remain stable，proving that this scheme has a practical application significance.

【Key words】Federated Transfer Learning（FTL）；secure multiparty computation；secret sharing；Oblivious Transfer（OT）protocol；homomorphic encryption

DOI：10. 19678/j. issn. 1000-3428. 0064452

基金项目：上海市基础研究重点项目（21JC1400600）。

作者简介：郑云涛（1997—），男，硕士研究生，主研方向为多方安全计算、区块链、密码学；叶家炜，工程师、硕士。

收稿日期：2022-04-13　　**修回日期：**2022-05-16　　**E-mail：**ytzheng19@fudan.edu.cn

0　概述

近年来，由于数据的存储与处理效率的提高，海量数据参与机器学习模型的训练成为可能。利用不同来源的数据进行机器学习模型训练并生成预测模型能够获得更多的特征信息，从而提高模型的预测准确率。然而，现实中大多数数据分散在不同的组织中，由于受到法律等的约束，无法通过集成来进行共同的机器学习模型训练，如医院的患者信息、银行的用户账户信息等数据并不能得到公开共享，这使得整合来自不同来源的数据比较困难。因此，数据的隐私保护技术成为大数据时代机器学习不可或缺的技术手段。

安全多方计算是一种能够在保证计算参与方的数据不被泄露的情况下完成最终计算过程的隐私保护技术。目前，研究人员提出了多个实现安全多方计算的协议，包括利用同态加密来直接对数据进行加密的协议[1]、利用秘密共享实现数据不被泄露的Shamir秘密共享方案[2]、茫然传输（Oblivious Transfer，OT）协议[3]等，以及相关的秘密分享协议框架，如SPDZ协议[4]、ABY协议[5]等。

联邦学习利用安全多方计算技术来对机器学习模型中的数据进行保护，在不泄露用户数据隐私的情况下共享数据并构建机器学习模型，使得训练出的模型的预测准确率不会受到损失。目前联邦学习的隐私保护机制的实现有基于模型聚类[6-8]、基于同态加密[9-11]、基于差分隐私[12]以及基于秘密分享[13]等相关研究。

FATE联邦学习是一个成熟的、已投入使用的基于同态加密的联邦机器学习框架，文献[14]提出的FATE联邦迁移学习框架，通过近似多项式的方法来拟合逻辑回归函数，并用Paillier同态加密[15]的方式，在保证参与联邦学习双方的数据隐私的情况下完成损失函数和梯度的计算，最终达到收敛完成模型的训练。然而，由于在实际的模型训练过程中需要对大量的数据进行加密处理和计算，模型训练的效率较低。文献[16]提出了通过批量加密的方法来提高同态加密的通信效率，但是本质上没有解决同态加密乘法的计算开销问题。

针对FATE联邦迁移学习（Federated Transfer Learning，FTL）中同态加密带来的计算开销的问题，本文提出一个基于OT协议的隐私保护两方矩阵运算方案，并拓展应用到FATE联邦迁移学习的设计中，在保证参与计算双方的数据隐私的前提下完成逻辑回归模型的损失函数和梯度计算。同时，针对OT协议带来的通信开销问题，通过OT扩展（OT extension）协议和批量传输进行优化，实现基于OT协议矩阵运算的FATE联邦迁移学习方案。

1　预备知识

1.1　联邦迁移学习

联邦学习最早由谷歌在2016年提出[17]，目的是通过分布在不同设备上的数据集来协同构建机器学习模型，并防止各个设备的数据泄露。

假设有n个数据持有组织$\{p_1,p_2,\cdots,p_n\}$，每个组织拥有自己的数据$D_i(1\leqslant i\leqslant n)$，通过中间数据的交互来共同训练机器学习模型$M_{\text{fed}}$。一种常规的联合机器学习方法是将所有组织的数据聚合在一起，共同训练模型得到模型M_{sum}。与普通的联合机器学习不同，联邦学习各个组织的数据D_i不会泄露给其他组织。根据V_{fed}和V_{sum}联邦学习与联合学习的准确性，假如$|V_{\text{fed}}-V_{\text{sum}}|<\delta$，则称该联邦学习算法有$\delta$-acc级损失。

联邦学习的分类有横向联邦学习、纵向联邦学习与联邦迁移学习。横向联邦学习适用于样本不同的情况下参与方数据集的特征空间相同的情况；纵向联邦学习适用于参与方样本空间重合度高，但各自的特征空间不同的情况；如果参与方之间的数据的特征空间和样本重合度都很低，则需要用到联邦迁移学习，如图1所示。

图1　联邦迁移学习框架

Fig.1　Framework of federated transfer learning

1.2　OT协议

OT协议是重要的密码学原语之一，在实际应用中，往往需要许多简单的OT_2^1协议来完成高级协议的设计，在执行k次的OT_2^1协议（记为$k\text{OT}_2^1$）中，发送方S持有k对比特对(m_0^i,m_1^i)，其中$m_0^i,m_1^i\in\{0,1\}$，而接收方R持有一个k比特的选择向量$\boldsymbol{b}$，通过执行k次协议后，R获得了$m_{b[i]}^i$，但是无法得知其他的比特位信息，而S无法得知$\boldsymbol{b}$的信息，从而保证了隐私性与保密性。目前研究人员提出许多高效的OT协议方案，如在半诚实模型下通过公钥机制来实现的Naor-Pinkas OT协议[18]等。

OT extension协议利用少数的Base OT协议将原本大量的OT协议数量大幅减少，能够极大地降低计算量和通信量。IKNP OT extension协议[19]通过选定安全参数k，能够将m次OT_2^1协议降低到k次的Base OT协议。发送方准备m对秘密消息作为协议的输入，接收方准备$\{0,1\}^m$比特对消息作为协议的输入。在协议执行过程中，双方在预处理阶段执行k次的Base OT协议，发送方作为Base OT的接收者，接收方随机产生k比特对消息作为Base OT的发送者。执行完k次Base OT协议后，双方即可以在交互阶段进行m次的OT_2^1协议的交互，而每次OT_2^1协议

的执行只需要进行少数的异或运算。接收方最终会获得自己选择的m个消息。IKNP OT extension不仅能够减少OT协议的通信量,也加快了计算效率。

本文通过调用大量的OT_2^1协议来实现两方的矩阵乘法运算,并采用IKNP OT extension协议提高OT协议的运行效率。

1.3 逻辑回归的泰勒展开近似

在线性回归模型中,模型训练本质上是需要拟合出一条线,使得训练样本中的数据点尽可能多地靠近这条线,即$y=\boldsymbol{W}\cdot\boldsymbol{x}$,其中:$\boldsymbol{W}$是要训练的模型参数,它通过随机化一个初始值得到;$\boldsymbol{x}$为参与训练的样本数据,它可以是一个向量,也可以称为特征值;$\boldsymbol{x}$与$\boldsymbol{W}$的维度大小是相同的,y是一个单值,也可以称为标签。

为了能够学习到模型的参数$\boldsymbol{W}$,一般采用损失函数Loss和随机梯度下降的算法来更新每一轮的$\boldsymbol{W}$。Loss函数为均方误差,计算公式如下:

$$\mathrm{Loss}(\boldsymbol{W})=\frac{1}{2}\left(\boldsymbol{x}\cdot\boldsymbol{W}-y\right)^2 \tag{1}$$

$$\boldsymbol{W}_j:=\boldsymbol{W}_j-\alpha\left(\boldsymbol{x}\cdot\boldsymbol{W}-y\right)\boldsymbol{x}_j \tag{2}$$

无论是Loss函数还是梯度计算,式(1)、式(2)都只涉及乘法和加法运算,因此可以使用多方安全计算来直接运用到线性回归问题上,同态加密与秘密分享的方法都是可行的,然而对于逻辑回归等模型的损失函数的计算,往往需要用多项式近似的方法来模拟损失函数[20-22]。文献[11]利用二阶泰勒展开的方式近似模拟损失函数并给出了精度损失的比较。对于损失函数:$l_1(y,\varphi)=\log_a(1+\exp(-y\varphi))$,其中,$\varphi$为训练模型的预测值,其二阶泰勒展开式为:

$$l_1(y,\varphi)=l_1(y,0)+\frac{1}{2}C(y)\varphi+\frac{1}{8}D(y)\varphi^2 \tag{3}$$

其中:$C(y)=\left.\frac{\partial l_1}{\partial\varphi}\right|_{\varphi=0}$;$D(y)=\left.\frac{\partial^2 l_1}{\partial\varphi^2}\right|_{\varphi=0}$。

采用二阶的泰勒展开作为逻辑回归的损失函数拟合进行模型的训练。

2 基于OT协议的矩阵运算设计

设计一种基于OT协议的多方安全计算方案,并将其应用到矩阵乘法运算。

2.1 基于OT协议的整数乘法运算

将OT协议在比特位上的与运算扩展到简单的整数乘法运算,假设需要计算$a\cdot b$,其中,A持有a,B持有b,则:

1)A将a转换为二进制表示的形式,即$a=\sum_{i=0}^{m}a_i\cdot 2^i$,则:$a\cdot b=\sum_{i=0}^{m}a_i\cdot b\cdot 2^i$,对于每个$a_i\cdot b$可以调用$OT_2^1$协议。

2)B作为OT协议中的秘密提供方,随机化整数c,将c与$c+b$作为OT协议的输入。

3)A作为OT协议中的选择方,输入为a_i,输出结果为a_ib+c,即:当$a_i=0$时,输出为c;当$a_i=0$时,输出为$c+b$。

4)A将输出结果作为z_{a_i},$z_a=\sum_{i=0}^{m}z_{a\_i}\cdot 2^i$。

5)B将$-c$作为输出结果z_{b_i},$z_b=\sum_{i=0}^{m}z_{b\_i}\cdot 2^i$,则:$z_a+z_b=\sum_{i=0}^{m}a_i\cdot b\cdot 2^i=a\cdot b$,$z_a$、$z_b$即作为A与B关于$a\cdot b$的秘密分享。

2.2 基于OT extension协议的安全矩阵运算

构造基于OT extension协议的矩阵运算的多方安全计算方案。

对于矩阵$\boldsymbol{A}$和$\boldsymbol{B}$的运算,P_1拥有矩阵$\boldsymbol{A}$,P_2拥有矩阵$\boldsymbol{B}$,针对矩阵运算分别进行如下构造:

1)$\boldsymbol{A}+\boldsymbol{B}$

对于需要相加的所有原矩阵或秘密分享矩阵,P_1和P_2直接本地相加即可。

2)$\boldsymbol{A}\times\boldsymbol{B}$(矩阵$\boldsymbol{A}$与$\boldsymbol{B}$的按位运算)

P_1和P_2对于矩阵中的每对元素乘法分别执行基于OT协议的乘法运算,并对所有的OT_2^1协议通过OT extension进行优化。

3)$\boldsymbol{a}*\boldsymbol{B}$(矩阵的数乘)

$\boldsymbol{A}\times\boldsymbol{B}$的特殊形式,但是由于每次乘法的都是同一个数,相当于每次OT_2^1协议的选择都一致,因此通过算法1来优化OT extension的输入。P_1通过调用Batch($\boldsymbol{C}$)与Batch($\boldsymbol{C}+\boldsymbol{B}$)得到OTInput<$\boldsymbol{C}$>,OTInput<$\boldsymbol{C}+\boldsymbol{B}$>作为OT extension的输入,$P_2$转化整数$a=\sum_{i=0}^{m}a_i\cdot 2^i$,将$c_{\mathrm{choice}}=\cup\{a_i\}$作为OT extension的输入。$P_2$将收到的OTOutput<$\boldsymbol{C}$>或OTOutput<$\boldsymbol{C}+\boldsymbol{B}$>反向解码回矩阵,执行基于OT协议的整数乘法运算。

算法1 OT协议批处理算法Batch

输入 矩阵$\boldsymbol{M}$

输出 合并序列OTInput<$\boldsymbol{M}$>

```
1.compute len = M.data.size();//记录矩阵元素的比特位
2.for each i in M:
3.      while(I <OT.maxsize())://当合并的数小于
//OT extension输入的上限时
4.              compute:I = (I≪len) + i;
5.      end
6.      OTInput.push_back(I);
7.end
```

4)$\boldsymbol{B}\cdot\boldsymbol{A}$(矩阵不具有交换律)

$\boldsymbol{B}\cdot\boldsymbol{A}$矩阵运算有以下2个阶段:

(1)初始化阶段

① P_1将矩阵$\boldsymbol{A}_{n\times l}$转换为二进制表达的形式:

$$\boldsymbol{A}=\sum_{i,j}a_{ij}\cdot\boldsymbol{E}_{ij}=\sum_{i,j}\left(\sum_{k=0}^{m}a_{ijk}\cdot 2^k\right)\cdot\boldsymbol{E}_{ij}$$

在上式中，m为矩阵数的比特长度，$\boldsymbol{E}_{ij}$定义如下：

$$\boldsymbol{E}[p][q]=\begin{cases}1, & p=i, q=j\\ 0, & \text{其他}\end{cases}$$

P_1构造长为$\boldsymbol{M}=n\times l\times m$的选择向量$\boldsymbol{c}_{\text{choice}}=\cup\{a_{ijk}\}$作为OT extension协议的输入。

②P_2根据矩阵整数的比特长度m与矩阵$\boldsymbol{A}_{n\times l}$的大小生成相应的随机矩阵组$\boldsymbol{C}_r$，$1\leqslant r\leqslant m\times n\times l$。

③P_2将$s_{\text{seceret}}=\{(\boldsymbol{C}_r,\boldsymbol{C}_r+\boldsymbol{B})\}$作为OT extension协议的输入。由于对于每个矩阵内元素的OT_2^1协议选择都是一致的，因此可以通过运行算法1，将矩阵中的元素进行合并得到OTInput<$\boldsymbol{C}_r$>，OTInput<$\boldsymbol{C}_r+\boldsymbol{B}$>输入到OT extension框架中，以此减少需要进行的OT_2^1协议数量，提升OT extension的运行效率。

（2）交互阶段

①双方根据输入运行OT extension协议，得到各自的输出。

② P_1作为协议的输入方，得到批处理的矩阵输出OTOutput$<\boldsymbol{C}_r>$，OTOutput$<\boldsymbol{C}_r+\boldsymbol{B}>$反向解码回矩阵得到$R_{\text{Received}}$：

$$R_{\text{Received}}=\begin{cases}\boldsymbol{C}_r, & a_{ijk}=0\\ \boldsymbol{C}_r+\boldsymbol{B}, & a_{ijk}=1\end{cases}$$

③P_1将自己的秘密分享设置为：

$$s_0=\sum_{i,j}\left(\sum_{k=0}^{m}R_{\text{Received}}\cdot 2^k\right)\cdot\boldsymbol{E}_{ij}$$

$$s_1=\sum_{i,j}\left(\sum_{k=0}^{m}(-C_r)\cdot 2^k\right)\cdot\boldsymbol{E}_{ij}$$

④$\boldsymbol{B}\cdot\boldsymbol{A}=s_0+s_1$。

由于联邦学习中涉及大量矩阵运算，本文提出的基于OT协议的矩阵算法不仅能够大幅减少OT协议执行的数目，而且通过OT extension将所要处理的OT协议数目进一步减少，从而提高了数据的运行效率。

3　基于OT矩阵乘法的FTL方案设计

如图2所示，模型包括两个参与方A与B，共同进行机器学习的模型训练，每个参与方的样本数量为n。主要由以下3个部分组成：

1）参与方A拥有一定数量n的样本，每个样本只有对应的特征值$X_{ai}=(x_{ai1},x_{ai2},\cdots,x_{ail})$和标签$Y_{ai}$，$(X_{ai},Y_{ai})\in D_{\text{A}}$。参与方B拥有一定数量$m$的样本，每个样本只有对应的特征值$X_{bi}=(x_{bi1},x_{bi2},\cdots,x_{bil})$，$X_{bi}\in D_{\text{B}}$。$D_{\text{A}}$与$D_{\text{B}}$有少量的重合样本$D_{\text{C}}$。假设A、B事先知道重合的样本ID，否则可以用RSA加密机制对样本ID进行盲化，然后进行样本对齐[6]。

2）A和B各自拥有的机器学习模型训练服务器S1、S2，这两个服务器各自受控于A与B，不能进行共谋攻击。该服务器只负责机器学习模型相关的计算，如特征值计算、梯度计算、损失函数计算等。

3）OT协议矩阵运算框架负责对计算出来的特征值进行信息的批量处理，并进行矩阵的安全计算。

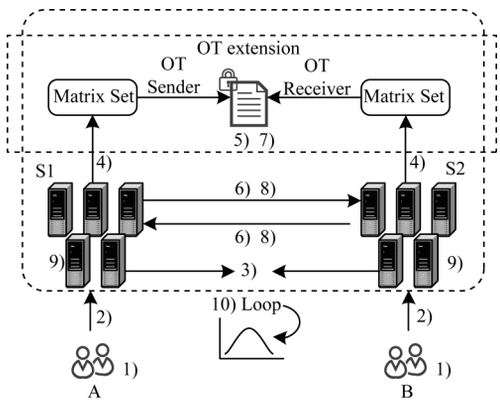

图2　基于OT协议联邦迁移学习框架

Fig.2　Framework of federated transfer learning based on OT protocol

假设参与方之间的通信存在安全通信信道，且双方的消息具有可验证性，图2中的步骤标识对应流程如下：

1）A与B作为机器学习模型的参与方，在每一轮迭代中初始化各自模型的参数W_{A}、W_{B}。

2）A与B将本地的样本数据和初始化参数传输到机器模型训练服务器S1与S2。

3）S1与S2通过RSA加密机制盲化样本ID，进行样本对齐（可选）。

4）A与B用各自计算模型的特征值$\boldsymbol{u}^{\text{A}}$、$\boldsymbol{u}^{\text{B}}$，输入OT协议框架准备进行安全矩阵运算。

5）A与B通过OT协议的秘密共享方案来共同计算Loss值，分别得到Loss的共享值$\langle\text{Loss}\rangle_{\text{A}}$与$\langle\text{Loss}\rangle_{\text{B}}$。

6）B将$\langle\text{Loss}\rangle_{\text{B}}$发送给A，A得以计算出整体的Loss值判断是否收敛。

7）A和B共同计算$\dfrac{\partial L}{\partial W_{\text{A}}}$和$\dfrac{\partial L}{\partial W_{\text{B}}}$，并获得$\left\langle\dfrac{\partial L}{\partial W_{\text{A}}}\right\rangle_{\text{A}}$、$\left\langle\dfrac{\partial L}{\partial W_{\text{A}}}\right\rangle_{\text{B}}$和$\left\langle\dfrac{\partial L}{\partial W_{\text{B}}}\right\rangle_{\text{A}}$、$\left\langle\dfrac{\partial L}{\partial W_{\text{B}}}\right\rangle_{\text{B}}$。

8）A将$\left\langle\dfrac{\partial L}{\partial W_{\text{B}}}\right\rangle_{\text{A}}$发送给B，B将$\left\langle\dfrac{\partial L}{\partial W_{\text{A}}}\right\rangle_{\text{B}}$发送给A。

9）A通过计算得到$\dfrac{\partial L}{\partial W_{\text{A}}}$，更新梯度；B通过计算得到$\dfrac{\partial L}{\partial W_{\text{B}}}$，更新梯度。

10）返回步骤4），直到模型收敛。

定义1（安全性假设）　当A、B参与方都是半诚实模型时，任何恶意敌手可能且仅可能控制其中一方，则对M有$(O_{\text{A}},O_{\text{B}})=\text{M}(I_{\text{A}},I_{\text{B}})$，其中，$I_{\text{B}}$、$O_{\text{B}}$是B的输入和输出结果，$I_{\text{A}}$、$O_{\text{A}}$是A的输入和输出结果。假设A是被恶意方控制的，如果对于任意数量的$(I'_{\text{B}},O'_{\text{B}})$，都有对应的$(O_{\text{A}},O'_{\text{B}})=\text{M}(I_{\text{A}},I'_{\text{B}})$，即恶意敌手无法分辨输出结果与随机结果，则该方案是安全的。

3.1 系统初始化阶段

假设OT extension的安全参数为κ。

$\text{Setup}(W_{\text{A}}, W_{\text{B}}, \kappa) \to (\cup \boldsymbol{r}, \cup \boldsymbol{s}, \cup \boldsymbol{K})$

1)服务器S1、S2根据输入的模型参数确定矩阵运算的维数,确定OT extension需要进行的OT_2^1协议数量:$m \leftarrow (W_{\text{A}}, W_{\text{B}})$。

2)服务器S2作为OT extension协议的接收方,对于每个待计算矩阵,产生m维向量$\boldsymbol{r}=(r_1, r_2, \cdots, r_m)$,$r_j \in \{0,1\}$,$1 \leqslant j \leqslant m$,作为自己的选择,初始化向量$\boldsymbol{r}$的集合$\cup \boldsymbol{r}$。

3)选择Base OT的数目κ作为安全参数。

4)发送方产生κ个随机数作为Base OT的选择消息,$\boldsymbol{s}=(s_1, s_2, \cdots, s_\kappa)$,$s_i \in \{0,1\}$,$1 \leqslant i \leqslant \kappa$,初始化向量$\boldsymbol{s}$的集合$\cup \boldsymbol{s}$。

5)接收方产生κ对随机数作为Base OT的发送消息:$\boldsymbol{K}=\{(K_i^0, K_i^1)\}_{i=1}^{\kappa}$,初始化向量$\boldsymbol{K}$的集合$\cup \boldsymbol{K}$。

3.2 模型训练阶段-计算损失函数

$\text{Train}_1(W_{\text{A}}, W_{\text{B}}, D_{\text{A}}, D_{\text{B}}, D_{\text{C}}) \to L$:

1)A和B对于每个样本i,将其输入到模型中计算出特征值:$\boldsymbol{u}_i^{\text{A}} \leftarrow \text{Net}^{\text{A}}(W_{\text{A}}, D_{\text{A}})$,$\boldsymbol{u}_i^{\text{B}} \leftarrow \text{Net}^{\text{B}}(W_{\text{B}}, D_{\text{B}})$。样本集合的特征值集合矩阵为$\boldsymbol{u}^{\text{A}}$、$\boldsymbol{u}^{\text{B}}$。

2)对于逻辑回归的损失函数计算,根据式(3)有:

$$L=\sum_i^{|D_{\text{C}}|}\left(l_1\left(Y_{ai},0\right)+\frac{1}{2}\boldsymbol{C}(Y_{ai})\boldsymbol{\Phi}_{\text{A}}(\boldsymbol{u}_i^{\text{B}})'+\frac{1}{8}\boldsymbol{D}(Y_{ai})\boldsymbol{\Phi}_{\text{A}}((\boldsymbol{u}_i^{\text{B}})')'(\boldsymbol{u}_i^{\text{B}}))\boldsymbol{\Phi}_{\text{A}}'\right) \tag{4}$$

其中:$\boldsymbol{\Phi}_{\text{A}}=\frac{1}{|D_{\text{A}}|}\sum_1^{|D_{\text{A}}|}Y_{ai}\boldsymbol{u}_i^{\text{A}}$;$\boldsymbol{C}\left(Y_{ai}\right)=Y_{ai}$;$\boldsymbol{D}\left(Y_{ai}\right)=Y_{ai}^2$。

3)服务器S1将计算得到的$\frac{1}{2}\boldsymbol{C}\left(Y_{ai}\right)\boldsymbol{\Phi}_{\text{A}}$矩阵、$\frac{1}{8}\boldsymbol{D}(Y_{ai})\boldsymbol{\Phi}_{\text{A}}$和$\boldsymbol{\Phi}_{\text{A}}'$矩阵输入到OT协议框架矩阵集合中,S2将计算得到的$(\boldsymbol{u}_i^{\text{B}})'$和$((\boldsymbol{u}_i^{\text{B}})')'$向量输入到矩阵集合框架中。

4)通过第2节中的OT矩阵乘法分别计算,得到:

$$\left\langle M_{1i}\right\rangle_{\text{A}}^{L}+\left\langle M_{1i}\right\rangle_{\text{B}}^{L}=\frac{1}{2}\boldsymbol{C}\left(Y_{ai}\right)\boldsymbol{\Phi}_{\text{A}}(\boldsymbol{u}_i^{\text{B}})'$$

$$\left\langle M_{2i}\right\rangle_{\text{A}}^{L}+\left\langle M_{2i}\right\rangle_{\text{B}}^{L}=\frac{1}{8}\boldsymbol{D}(Y_{ai})\boldsymbol{\Phi}_{\text{A}}((\boldsymbol{u}_i^{\text{B}})')'(\boldsymbol{u}_i^{\text{B}})'$$

$$\left\langle M_{3i}\right\rangle_{\text{A}}^{L}+\left\langle M_{3i}\right\rangle_{\text{B}}^{L}=\left\langle M_{2i}\right\rangle_{\text{B}}^{L}\boldsymbol{\Phi}_{\text{A}}'$$

$$\left\langle M_{5i}\right\rangle_{\text{A}}^{L}+\left\langle M_{5i}\right\rangle_{\text{B}}^{L}=\left\langle M_{2i}\right\rangle_{\text{B}}\left\langle M_{3i}\right\rangle_{\text{A}}$$

则有:

$$\langle\text{Loss}\rangle_{\text{A}}=\sum_i^{|D_{\text{C}}|}\left(l_1(Y_{ai},0)+\left\langle M_{1i}\right\rangle_{\text{A}}^{L}+\left\langle M_{2i}\right\rangle_{\text{A}}^{L}\boldsymbol{\Phi}_{\text{A}}'+\left\langle M_{3i}\right\rangle_{\text{A}}^{L}\right)$$

$$\langle\text{Loss}\rangle_{\text{B}}=\sum_i^{|D_{\text{C}}|}\left(\left\langle M_{1i}\right\rangle_{\text{B}}^{L}+\left\langle M_{3i}\right\rangle_{\text{B}}^{L}\right)$$

$$L=\langle\text{Loss}\rangle_{\text{A}}+\langle\text{Loss}\rangle_{\text{B}}$$

5)B将$\langle\text{Loss}\rangle_{\text{B}}$发送给A,A得以计算出整体的$\mathcal{L}$值来判断是否收敛;如果收敛,则模型训练结束。输出相关参数W_{A}、W_{B}。

3.3 模型训练阶段-计算梯度

$$\text{Train}_2(\boldsymbol{u}^{\text{A}}, \boldsymbol{u}^{\text{B}}, L) \to \left(\frac{\partial L}{\partial W_{\text{A}}}, \frac{\partial L}{\partial W_{\text{B}}}\right)$$

1)假设损失函数$\mathcal{L}$值并没有收敛,则需要计算对应的梯度值,根据式(4)得到:

$$\frac{\partial L}{\partial W_{\text{A}}}=\sum_j^{|D_{\text{A}}|}\sum_i^{|D_{\text{C}}|}\left(\frac{1}{4}D\left(Y_{ai}\right)\boldsymbol{\Phi}^{\text{A}}\left(\left(\boldsymbol{u}_i^{\text{B}}\right)'\right)'\left(\boldsymbol{u}_i^{\text{B}}\right)'+\frac{1}{2}\boldsymbol{C}(Y_{ai})\left(\boldsymbol{u}_i^{\text{B}}\right)'\frac{\partial\boldsymbol{\Phi}^{\text{A}}}{\partial\boldsymbol{u}_j^{\text{A}}}\times\frac{\partial u_j^{\text{A}}}{\partial W_{\text{A}}}\right)$$

$$\frac{\partial L}{\partial W_{\text{B}}}=\sum_i^{|D_{\text{C}}|}\left(\frac{1}{8}\times\frac{\partial\left(\left(\boldsymbol{u}_i^{\text{B}}\right)'\right)'\left(\boldsymbol{u}_i^{\text{B}}\right)'}{\partial\boldsymbol{u}_i^{\text{B}}}\boldsymbol{D}(Y_{ai})(\boldsymbol{\Phi}^{\text{A}})'\boldsymbol{\Phi}^{\text{A}}\frac{\partial u_i^{\text{B}}}{\partial W_{\text{B}}}\right)+\sum_i^{|D_{\text{C}}|}\left(\frac{1}{2}C(Y_{ai})\boldsymbol{\Phi}^{\text{A}}\frac{\partial\left(\boldsymbol{u}_i^{\text{B}}\right)'}{\partial\boldsymbol{u}_i^{\text{B}}}\times\frac{\partial\boldsymbol{u}_i^{\text{B}}}{\partial W_{\text{B}}}\right)$$

2)同理,A和B通过OT矩阵运算共同计算$\frac{\partial L}{\partial W_{\text{A}}}$和$\frac{\partial L}{\partial W_{\text{B}}}$,并获得各自的分享$\left\langle\frac{\partial L}{\partial W_{\text{A}}}\right\rangle_{\text{A}}$、$\left\langle\frac{\partial L}{\partial W_{\text{A}}}\right\rangle_{\text{B}}$和$\left\langle\frac{\partial L}{\partial W_{B}}\right\rangle_{\text{A}}$、$\left\langle\frac{\partial L}{\partial W_{B}}\right\rangle_{\text{B}}$:

$$\left\langle M_{1i}\right\rangle_{\text{A}}^{g^{\text{A}}}+\left\langle M_{1i}\right\rangle_{\text{B}}^{g^{\text{A}}}=\frac{1}{4}D\left(Y_{ai}\right)\boldsymbol{\Phi}^{\text{A}}\left(\left(\boldsymbol{u}_i^{\text{B}}\right)'\right)'\left(\boldsymbol{u}_i^{\text{B}}\right)'$$

$$\left\langle M_{2i}\right\rangle_{\text{A}}^{g^{\text{A}}}+\left\langle M_{2i}\right\rangle_{\text{B}}^{g^{\text{A}}}=\frac{1}{2}C(Y_{ai})\left(\boldsymbol{u}_i^{\text{B}}\right)'$$

$$\left\langle M_{3i}\right\rangle_{\text{A}}^{g^{\text{A}}}+\left\langle M_{3i}\right\rangle_{\text{B}}^{g^{\text{A}}}=\left\langle M_{2i}\right\rangle_{\text{B}}^{g^{\text{A}}}\frac{\partial\boldsymbol{\Phi}^{\text{A}}}{\partial\boldsymbol{u}_j^{\text{A}}}\frac{\partial\boldsymbol{u}_j^{\text{A}}}{\partial W_{\text{A}}}$$

计算得到梯度相关的分享值:

$$\left\langle\frac{\partial L}{\partial W_{\text{A}}}\right\rangle_{\text{A}}=\sum_j^{|D_{\text{A}}|}\sum_i^{|D_{\text{C}}|}\left(\left\langle M_{1i}\right\rangle_{\text{A}}^{g^{\text{A}}}+\left\langle M_{2i}\right\rangle_{\text{A}}^{g^{\text{A}}}\frac{\partial\boldsymbol{\Phi}^{\text{A}}}{\partial\boldsymbol{u}_j^{\text{A}}}\frac{\partial\boldsymbol{u}_j^{\text{A}}}{\partial W_{\text{A}}}+\left\langle M_{3i}\right\rangle_{\text{A}}^{g^{\text{A}}}\right)$$

$$\left\langle\frac{\partial L}{\partial W_{\text{A}}}\right\rangle_{\text{B}}=\sum_j^{|D_{\text{A}}|}\sum_i^{|D_{\text{C}}|}\left(\left\langle M_{1i}\right\rangle_{\text{B}}^{g^{\text{A}}}+\left\langle M_{3i}\right\rangle_{\text{B}}^{g^{\text{A}}}\right)$$

同理,可计算获得$\left\langle\frac{\partial L}{\partial W_{\text{B}}}\right\rangle_{\text{A}}$与$\left\langle\frac{\partial L}{\partial W_{\text{B}}}\right\rangle_{\text{B}}$。

3)A将$\left\langle\frac{\partial L}{\partial W_{\text{B}}}\right\rangle_{\text{A}}$发送给B,B将$\left\langle\frac{\partial L}{\partial W_{\text{A}}}\right\rangle_{\text{B}}$发送给A。

4)A计算得到$\frac{\partial L}{\partial W_{\text{A}}}=\left\langle\frac{\partial L}{\partial W_{\text{A}}}\right\rangle_{\text{A}}+\left\langle\frac{\partial L}{\partial W_{\text{A}}}\right\rangle_{\text{B}}$,更新梯度;B计算得到$\frac{\partial L}{\partial W_{\text{B}}}=\left\langle\frac{\partial L}{\partial W_{\text{B}}}\right\rangle_{\text{A}}+\left\langle\frac{\partial L}{\partial W_{\text{A}}}\right\rangle_{\text{B}}$,回到3.2节步骤2,重新计算损失函数。

4 安全性性能分析

4.1 安全性

定理1 在半诚实模型假设下,两方的OT矩阵运算方案是安全的。

证明 在半诚实模型的条件下,参与矩阵运算的双方P_1、P_2是遵守协议规则的,但尝试去获取更多

相关的信息。假设存在某敌手Adversary，每次最多只能控制P_1、P_2其中的一个。不失一般性地，假设对于矩阵的乘法$\boldsymbol{B}\cdot\boldsymbol{A}$：当Adversary控制$\boldsymbol{A}$持有者作为OT协议的接收方时，发送方每次执行OT协议都会根据矩阵整数的比特长度m与矩阵$\boldsymbol{A}_{n\times l}$的大小生成相应的随机矩阵组$\boldsymbol{C}_r$，$1\leqslant r\leqslant m\times n\times l$，以$\boldsymbol{C}_r+\boldsymbol{B}$与$\boldsymbol{C}_r$作为输入，敌手或另一方服务器看来都是随机值的矩阵对，不会暴露任何有效信息；当Adversary控制$\boldsymbol{B}$持有者作为OT协议的发送方时，选择方的选择隐私性依赖于OT协议的安全性，在半诚实模型下是安全的，不会暴露自己的任何比特位信息。矩阵的按位乘法和数乘都将转化为基于OT的乘法运算。同理，发送方每次执行OT协议都会对应地产生随机数c，以c和$c+b$作为输入，在攻击者看来都是无法分辨的随机数。因此，对于任意一个矩阵运算，对于诚实方的任意Input，另一方接收到的Output都是与随机值不可区分的，因此基于OT协议的两方矩阵运算是安全的。

4.2　隐私性

定理2　在半诚实模型安全性假设下，基于OT协议的联邦迁移学习方案不会暴露任何参与方的隐私信息。

证明　在半诚实模型假设下，由于关于损失函数与梯度的计算都是基于OT协议的矩阵运算构造方案，在矩阵运算的过程中不会暴露任何参与运算的数据的信息，因此可以推出：首先，对于恶意敌手Adversary，其在所有矩阵相关的运算中无法获得原始矩阵的任何信息；其次，对于最终传输的结果$\langle\mathrm{Loss}\rangle_{\mathrm{B}}$、$\left\langle\frac{\partial L}{\partial W_{\mathrm{B}}}\right\rangle_{\mathrm{A}}$和$\left\langle\frac{\partial L}{\partial W_{\mathrm{A}}}\right\rangle_{\mathrm{B}}$都是经过运算后的中间结果，并不会对A与B的数据特征信息$\boldsymbol{u}^{\mathrm{A}}$与$\boldsymbol{u}^{\mathrm{B}}$造成泄露；最后，这些中间结果都是由不同矩阵乘法或加法得到的累加值，在外部看来只是1个随机的值，而不是某2个矩阵直接相加或相乘得到的结果，即使在A或B重建恢复出这些中间结果之后，也不能反推出相应的模型参数值或数据特征信息。

在半诚实模型下的每一轮迭代中，A或B只能学习得到自己对应的梯度值或损失函数值，以此更新自己的参数，即对于任意数量的诚实方的输入，Adversary无法得到与诚实方的输入相关的任何信息。

4.3　可扩展性

FATE联邦迁移学习方案在效率优化和应用上都有很好的扩展性。

首先，在矩阵运算效率的提升方面，构造了线上的基于OT协议矩阵乘法运算，可以对应地在线下构造大量的乘法三元组来进行优化[23-24]，构造乘法三元组的过程可以作为预处理输入，从而增加线上的效率，具有很高的实际应用意义。

其次，对于应用上的扩展，构造了逻辑回归机器学习模型的训练方案，而简单神经网络的每一层的激活函数与逻辑回归类似，在二分类的问题上，可以拓展应用到神经网络的拟合中，从而实现更高精确度的机器模型训练方案[13,25]。

4.4　效率分析

在局域网环境下基于FATE联邦迁移学习框架代码进行了实现，运行机器是Intel 5220R 2.2 GHz的CPU处理器，为24核48线程以及128 GB RECC DDR4内存，环境为64位的Ubuntu20.04。实验使用C++语言编写IKNP OT extension库并通过批量处理优化效率，通过extern C关键字由C语言编译成动态库libOTE.so供机器模型训练代码使用。同时，逻辑回归模型训练采用Python语言以及numpy库进行编写。

表1所示为本文方案与基于同态加密的逻辑回归联邦迁移学习（FATE FTL）方案训练效率比较，实验模拟方案的样本数量为500个，样本重合数量为50个，特征维度为32个，IKNP OT extension的安全参数设置为$\kappa=1\,024$。迭代次数与收敛时间取10次实验的平均值。可以看出：在同为2个参与方的网络架构和相同的机器模型训练下，本文方案的平均迭代次数为862次，优于FATE联邦迁移学习方案的945次。本文方案的逻辑回归模型训练平均收敛时间为13 582.738 5 s，而基于同态加密的FATE联邦迁移学习的模型训练平均收敛时间为18 474.326 1 s，本文方案要比FATE联邦迁移学习方案快大约25%。另一方面，本文方案可以做乘法三元组的优化，将方案转化成线下部分和线上部分，通过预处理在线下生成大量的乘法三元组供线上使用，从而提高线上运行效率，可扩展性更高。

表1　本文方案与FATE联邦迁移学习方案性能对比

Table 1　Performance comparison between proposed scheme and FATE federated transfer learning scheme

性能参数	本文方案	FATE FTL方案
架构	两方	两方
机器学习模型	逻辑回归	逻辑回归
密码学支持	IKNP OT extension	Paillier同态加密
神经网络拓展	是	是
线下乘法三元组拓展	是	否
迭代次数	862	945
收敛时间/s	13 582.738 5	18 474.326 1

图3所示为本文方案在不同特征维度下的收敛时间变化与同态加密方案的联邦迁移学习方案的比较，实验模拟方案的样本数量为500个，样本重合数量为50个。可以看出：在特征维度增加的条件下，本文方案的时间开销呈线性稳定增长，依然拥有较好的性能稳定性，且平均效率比基于同态加密的FATE联邦迁移学习方案高25%左右。可见本文方案在较为复杂的样本类型中仍然具备较好的性能，具有一定的实际应用意义。

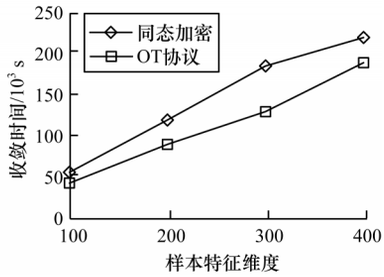

图3 不同特征维度下的联邦迁移学习收敛时间

Fig.3 Convergence time of federated transfer learning under different feature dimensions

5 结束语

本文提出一种基于茫然传输协议的安全矩阵计算方案。通过实现矩阵的加法、乘法与数乘的安全运算,完成逻辑回归模型的损失函数与梯度更新的计算,并将其嵌入到FATE联邦迁移学习的框架中。同时,通过OT extension技术和通信批处理计算,减少矩阵运算所需的OT协议的通信开销。实验结果表明,与同态加密的方案相比,本文方案能够有效提高FATE联邦迁移学习框架中模型的收敛效率。下一步将研究拓展本文方案在多方机器学习模型上的训练,以及通过乘法三元组结构来进行线下的预处理,从而提高线上的效率。

参考文献

[1] GENTRY C, HALEVI S. Implementing gentry's fully-homomorphic encryption scheme [C]//Proceedings of Annual International Conference on the Theory and Applications of Cryptographic Techniques. Berlin, Germany: Springer, 2011: 129-148.

[2] SHAMIR A. How to share a secret[J]. Communications of the ACM, 1979, 22(11): 612-613.

[3] RABIN M O. How to exchange secrets by oblivious transfer: TR-81 [R]. Aiken Computation Laboratory, Harvard University, 1981.

[4] DAMGÅRD I, PASTRO V, SMART N, et al. Multiparty computation from somewhat homomorphic encryption[C]// Proceedings of CRYPTO'12. Berlin, Germany: Springer, 2012: 643-662.

[5] DEMMLER D, SCHNEIDER T, ZOHNER M. ABY-A framework for efficient mixed-protocol secure two-party computation [C]//Proceedings of 2015 Network and Distributed System Security Symposium. Washington D. C., USA: IEEE Press, 2015: 257-268.

[6] MCMAHAN B, MOORE E, RAMAGE D, et al. Communication-efficient learning of deep networks from decentralized data[C]// Proceedings of IEEE PMLR'17. Washington D. C., USA: IEEE Press, 2017: 1273-1282.

[7] PAPERNOT N, ABADI M, ERLINGSSON Ú, et al. Semi-supervised knowledge transfer for deep learning from private training data[EB/OL]. [2022-03-10]. https://arxiv.org/abs/1610.05755.

[8] YUROCHKIN M, AGARWAL M, GHOSH S, et al. Bayesian nonparametric federated learning of neural networks[EB/OL]. [2022-03-10]. https://arxiv.org/abs/1905.12022.

[9] YANG T, ANDREW G, EICHNER H, et al. Applied federated learning: improving google keyboard query suggestions[EB/OL]. [2022-03-10]. https://arxiv.org/abs/1812.02903.

[10] HARDY S, HENECKA W, IVEY-LAW H, et al. Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption[EB/OL]. [2022-03-10]. https://arxiv.org/abs/1711.10677.

[11] SHARMA S, XING C P, LIU Y, et al. Secure and efficient federated transfer learning[C]//Proceedings of 2019 IEEE International Conference on Big Data. Washington D. C., USA: IEEE Press, 2019: 2569-2576.

[12] GEYER R C, KLEIN T, NABI M. Differentially private federated learning: a client level perspective[EB/OL]. [2022-03-10]. https://arxiv.org/abs/1712.07557.

[13] MOHASSEL P, ZHANG Y P. SecureML: a system for scalable privacy-preserving machine learning [C]// Proceedings of IEEE Symposium on Security and Privacy. Washington D. C., USA: IEEE Press, 2017: 19-38.

[14] LIU Y, KANG Y, XING C P, et al. A secure federated transfer learning framework[J]. IEEE Intelligent Systems, 2020, 35(4): 70-82.

[15] PAILLIER P. Public-key cryptosystems based on composite degree residuosity classes[C]//Proceedings of International Conference on the Theory and Applications of Cryptographic Techniques. Berlin, Germany: Springer, 1999: 223-238.

[16] 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. Washington D. C., USA: IEEE Press, 2020: 493-506.

[17] KONEČNÝ J, MCMAHAN H B, RAMAGE D, et al. Federated optimization: distributed machine learning for on-device intelligence[EB/OL]. [2022-03-10]. https://arxiv.org/abs/1610.02527.

[18] NAOR M, PINKAS B. Efficient oblivious transfer protocols[C]// Proceedings of the 12th Annual ACM-SIAM Symposium on Discrete Algorithms. New York, USA: ACM Press, 2001: 448-457.

[19] ISHAI Y, KILIAN J, NISSIM K, et al. Extending oblivious transfers efficiently[C]//Proceedings of CRYPTO'03. Berlin, Germany: Springer, 2003: 145-161.

[20] AONO Y, HAYASHI T, PHONG L T, et al. Scalable and secure logistic regression via homomorphic encryption[C]// Proceedings of the 6th ACM Conference on Data and Application Security and Privacy. New York, USA: ACM Press, 2016: 142-144.

[21] PHONG L T, AONO Y, HAYASHI T, et al. Privacy-preserving deep learning via additively homomorphic encryption[J]. IEEE Transactions on Information Forensics and Security, 2018, 13(5): 1333-1345.

[22] KIM A, SONG Y, KIM M, et al. Logistic regression model training based on the approximate homomorphic encryption[J]. BMC Medical Genomics, 2018, 11(4): 83.

[23] CHEN H, KIM M, RAZENSHTEYN I, et al. Maliciously secure matrix multiplication with applications to private deep learning[C]//Proceedings of ASIACRYPT'20. Berlin, Germany: Springer, 2020: 31-59.

[24] BOYLE E, COUTEAU G, GILBOA N, et al. Efficient pseudorandom correlation generators: silent OT extension and more [C]//Proceedings of CRYPTO' 19. Berlin, Germany: Springer, 2019: 489-518.

[25] BONAWITZ K, EICHNER H, GRIESKAMP W, et al. Towards federated learning at scale: system design[EB/OL]. [2022-03-10]. https://arxiv.org/abs/1902.01046.

编辑 索书志