基于揭序加密的联邦决策树安全比较协议

韩朝阳[1),3)] 葛春鹏[2)] 刘 哲[3)] 方黎明[1)]

[1)](南京航空航天大学计算机科学与技术学院 南京 210016)

[2)](山东大学软件学院 济南 250100)

[3)](之江实验室 杭州 311121)

摘 要 得益于联邦学习的发展,多个参与方得以越来越方便地在保护隐私的前提下协同训练多种机器学习模型.随着联邦学习训练方案的逐渐成熟,在这些模型上进行高效联邦预测任务的需求也日益受到关注.目前对联邦学习得到模型的应用的研究大都处于复用训练范式的范畴,而实际上这一做法造成了额外的开销,联邦预测任务存在着极大的效率提升空间.本文对基于树模型的外包预测服务进行研究.首先明确阐述了"联邦预测"任务,并指出设计高效解决方案面临着数据异构分布和用户数据动态变化的挑战.针对这些挑战,本文提出了新颖的安全比较协议 OREC. OREC 通过一个不可信的第三方的协助,使得两个参与方得以秘密地比较他们的私密值.所设计的协议基于揭序加密,其通过公开函数可以揭示出密文之间的顺序关系.为了增强其隐私性,我们进一步设计了一次一密机制并引入了混淆密码序列技巧.我们对协议进行了全面的安全性分析,并证明了其面对恶意第三方的鲁棒性.此外,我们还通过大量实验验证了所提出协议的高效性和可扩展性.

关键词 联邦学习;联邦预测;安全比较;揭序加密;外包计算;隐私计算

中图法分类号 TP18 **DOI号** 10.11897/SP.J.1016.2024.00892

A Secure Comparison Protocol for Federated Decision Trees Based on Order-Revealing Encryption

HAN Zhao-Yang[1),3)] GE Chun-Peng[2)] LIU Zhe[3)] FANG Li-Ming[1)]

[1)](*College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing* 210016)

[2)](*School of Software, Shandong University, Jinan* 250100)

[3)](*Zhejiang Lab, Hangzhou* 311121)

Abstract With the advancement of federated learning, multiple participants can collaboratively train diverse machine learning models while preserving privacy. As federated modeling techniques have been well-established, there is an increasing demand for efficient federated prediction tasks on these models. However, most existing research focuses on reusing training paradigms in federated learning, which results in additional overhead. In this paper, we specifically investigate outsourced prediction services for tree-based models, such as gradient boosting trees and random forests. To address this challenge, we propose the OREC secure comparison protocol. This protocol enables confidential values to be secretly compared with the assistance of an untrusted third party. By utilizing order-revealing encryption, the protocol unveils the order relationship between ciphertexts through public functions. To further enhance privacy, we introduce a one-time pad

收稿日期:2023-06-29;在线发布日期:2024-01-15. 本课题得到国家重点研发青年科学家项目(2021YFB3100700)、国家自然科学基金联合重点项目(U20B2049,U22B2030,U22B2029)、国家自然科学基金面上项目(62076125,62272228),江苏省基础研究杰出青年基金(BK20220075)资助.**韩朝阳**,博士研究生,主要研究领域为联邦学习、隐私计算. E-mail:sunrisehan@nuaa.edu.cn. **葛春鹏**,博士,教授,主要研究领域为可搜索加密、区块链. **刘 哲(通信作者)**,博士,教授,主要研究领域为密码工程、后量子密码. E-mail:zhe.liu@zhejianglab.com. **方黎明**,博士,教授,主要研究领域为可搜索加密、人工智能安全.

mechanism and obfuscated password sequences. Furthermore, we conduct a comprehensive security analysis to demonstrate the protocol's robustness against malicious third parties. Through extensive experiments, we validate its efficiency and scalability, highlighting its potential in practical applications.

Keywords　federated learning; federated inference; secure comparison; order-revealing encryption; outsourcing computation; privacy computing

1　引　言

在许多场景下,基于机器学习的外包数据分析已经得到广泛应用,例如在线诊断服务[1]、信用评估[2]等.在这些场景中,由于决策树模型具备较强的可解释性和优秀的模型性能,因此备受用户青睐.然而,由于隐私问题和相关数据法规(如 GDPR[3]和 CCPA[4])的制定,各方直接共享敏感数据进行模型预测的过程受到了限制.为了解决这一问题,研究者们提出了众多隐私计算框架,包括安全多方计算(Secure Multi-party Computation,MPC)和联邦学习(Federated Learning,FL)等.其中,联邦学习是一种实用的解决方案,能够协助多个用户联合训练机器学习模型并保护数据隐私.通过联邦学习算法,参与方可以在数据特征或用户分布异构的环境中训练多种模型,如决策树[5]、线性回归[6]和神经网络[7]等.

图 1 展示了在数据分布异构环境下的典型纵向联邦学习场景.参与者首先执行隐私集合交集协议(Private Set Intersection,PSI),以获取他们用户实体的交集.随后,他们以保护隐私的方式对齐数据并进行模型训练.最终,模型的所有者可以利用获取的模型提供模型预测服务.

在大多数情况下,模型的预测过程可以采用与隐私保护学习过程相同的方式进行.然而,直接将训练范式应用于预测过程会导致效率的降低,因为安全的联合建模算法,如利用同态加密[8]或安全多方计算[9]的方法,总是包含耗时的隐私保护模块.这种延迟问题对于基于树的模型尤为严重.尽管决策树的预测过程只需要比较操作,但联邦决策树的训练过程都包括了同态加密步骤[5,10-11],这是非常耗时的.此外,联邦学习算法很难处理只有一个模型提供者和一个数据所有者的情况,因为它们的隐私保护方法可能需要足够数量的参与方来实现安全(例如 ABY3[12]等安全多方计算框架).因此,迫切需要为

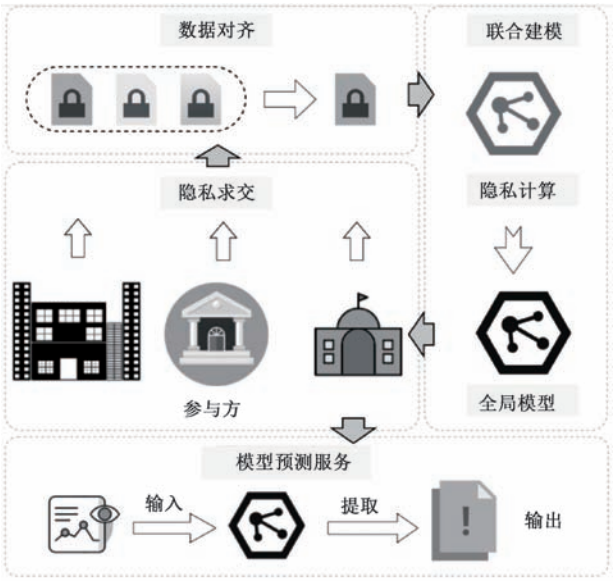

图 1　典型的纵向联邦学习场景

预测过程设计高效的隐私保护算法.先前的研究已经对决策树的安全预测过程进行了一些调查研究[1,13-16].然而,这些算法也未能充分利用基于树的模型的特点进行效率优化,并且没有考虑数据的异构性.即这些算法或是没有提供足够的隐私保护,或是具有大量的通信需求,使得整体效率依然低下.

对于基于决策树的模型来说,其预测流程为一系列的特征值与阈值的比较,因此其所需求的算子仅为安全比较操作,即若存在高效的安全比较协议(Secure Comparison Protocol)即可实现安全的外包决策树预测服务.安全比较协议是一种使得两个参与方能够在不向彼此透露实际数据的前提下比较他们的数据大小关系的 MPC 协议,其已经广泛研究应用于各种场景,包括私密集合交集,隐私保护机器学习和在线决策树预测服务等,这些场景都极其依赖数值的比较操作.安全比较协议已在多种情景下被广泛地研究,包括隐私求交[17]、隐私保护机器学习[18]和外包决策树预测[1,13-16,19]等.传统的安全比较协议构建方法通常依赖于安全多方计算技术,实现方法包括混淆电路[20]和秘密共享[21]等.另一

种思路则利用同态加密[8]来实现.这些方法的基本原理为在加密的待比较数据上进行减法计算,以获取两个值的差,并通过揭示该差的最高有效位(Most Significant Bit, MSB)来推导出最终的比较结果.这一技术已在联邦学习和隐私保护机器学习等领域得到了广泛的应用.然而,这些方法通常会带来较高的计算开销或过多的通信轮次,导致其在某些情况下因效率不足而变得不实用.为了提高效率,亦有研究者最近提出了一种基于可搜索对称加密(Searchable Symmetric Encryption, SSE)[22]的比较协议.然而,有研究者指出了基于 SSE 的方法安全性并未达标,其仍然面临着诸多隐私攻击的威胁[23-24].

除了上述提到的效率问题之外,之前的联邦决策树预测算法仍有应用场景窄的问题.先前的框架都是去中心化的设计,即所有参与方都会参与安全多方计算协议中的通信和计算,而这意味着具有有限的计算或通信资源的参与者难以应用这些协议完成联邦预测任务.在资源受限的情况下,参与方更倾向于选择将计算任务外包给第三方的隐私计算平台解决.随着近年来私有计算平台的快速增长,如谷歌机密计算、微软 Azure 机密计算、腾讯隐私计算等,这一需求场景已经变得越来越普遍.因此,一种支持将计算任务外包给第三方的安全计算协议也是亟需的.

为了解决上述问题,我们的目标是设计一个满足以下特点的安全比较协议:首先,它应具备计算效率高的特点,并仅对参与者有着极少量的通信需求;其次,该协议应提供强大的隐私保证,并能抵御多种隐私攻击;最后,该协议应能够在不可信任的第三方的协助下工作,并能够防御该第三方可能带来的潜在威胁.在本文中,我们会首先总结归纳模型外包预测的问题,并将其定义为"联邦预测"问题,即在机器学习模型已经获取的情况下,如何利用一个或多个数据所有者的数据进行隐私地预测.随后,基于对决策树模型特性的讨论和研究,我们在本工作中提出了基于揭序加密的安全比较协议(Order-revealing Encryption-based Comparison Protocol, OREC),其基于揭序加密(Order-revealing Encryption, ORE)实现了高效的安全比较,解决了现有协议的局限性,并可应用于决策树模型的联邦预测任务.所设计协议的核心在于统筹了加密算法和安全协议的优势.之前关于 ORE 的研究仅用于外包存储和安全数据库等场景,而在本文中,我们利用 ORE 的特性与优势,设计了轻量级的安全比较协议.为了进一步加强安全性,我们还将一次一密思想与和零知识证明(Zero Knowledge Proof, ZKP)[25]的验证技术原理也融入了所设计的协议中.这种整合最终导致了一个强大的安全比较协议的开发,它显著降低了通信开销,同时利用了 ORE 计算的能力.OREC 采用逐位计算来实现高效的执行效率,同时通过精心设计的通信协议来保持低通信开销.此外,如果恶意第三方试图异常行为,OREC 也可以立即识别,确保协议仍然安全和抵御此类攻击.我们在决策树模型的联邦预测任务场景中评估了 OREC,实验结果证明了其在效率方面相对于以前的解决方案具有很高的实用性和优越性.本文的贡献总结如下:

(1)正式描述了联邦预测问题,为其给出了详细定义并对其进行了符号化的描述.

(2)提出了 OREC,一种基于不可信第三方的新型安全比较协议,用于在两个参与方之间比较两个私密数值的大小关系.OREC 在计算和通信阶段都具有极高的效率,并提供了强大的隐私保证.

(3)对 OREC 的机密性和效率进行了形式化分析,证明了其单次运行仅具有极小的概率泄露 MSB.而所泄露的信息累积到可被攻击者利用的概率则几乎为 0.

(4)为了评估 OREC 的性能和安全性,我们进行了充足的实验,且实验结果证明了所提出算法的高效性与实用性.

2 相关工作

2.1 通用比较协议

目前大多数安全比较协议依赖于下述三种方法之一:同态加密、混淆电路(Garbled Circuits)或线性秘密分享(Linear Secret Sharing)[26].这些方法旨在验证两个数值之间的差是否大于零,并通过引入随机混淆值来保护该差的真实值[27].最新的基于混淆电路的方法,如 Obliv-C[28]和 TinyGarble2[29],将数学运算(如加法和减法)转化为了布尔电路.然后使用不经意传输(Oblivious Transfer)[30]和安全洗牌等加密技术处理这些电路,以确保每个参与方只能了解到自己输入的电路输出.然而,这些方法的计算开销和内存需求较高.基于秘密分享的协议,如 ABY3[12]、MPyC[31]和 SecureNN[32],使用 Shamir 的思想[33]来隐藏秘密值.它们将每个秘密值分成多个部分,分发给各参与方,然后在需要时使用重建算法将这些部分组合起来,实现共同计算差值的同时

保护隐私.然而,此类方法需要各方之间进行大量的消息交换,导致通信效率低下.面对如上这些协议,Veugen 等人[34]的研究曾指出,基于同态加密的安全比较协议在计算和通信开销方面具有更高的效率.由于安全比较不需要乘法和除法运算,因此可以使用较轻量级的偏同态加密方案,如 Paillier[35]和 CKKS[36].这些方法使用公钥加密秘密值,对密文进行计算,然后使用私钥解密结果,因此不会泄露任何中间信息.以上方法的低效性源于它们依赖的隐私保护策略旨在支持多种数值操作,而不仅仅是比较.因此,在只需要安全比较的情况下,使用这些方法会导致额外成本.

2.2　面向外包决策树的协议

近年来,针对基于决策树模型的外包预测服务,研究者们已提出了许多场景特化的安全比较协议.De Cock 等人[13]提出了一个通用的隐私模型预测框架,支持决策树,并利用遗忘输入选择协议来隐藏特征选择,利用对数轮次的安全比较协议[19]在决策树内部私密地排序数据.随后,Liang 等人[1]引入了安全决策树分类(SDTC)方案,使用 SSE 压缩决策树中的路径,以实现在线诊断系统的安全预测.然而,SDTC 暴露了数值精度,并具有较高的存储开销.此外,它只保护用户的数据隐私,可能泄漏有关模型的机密信息.为了解决这个限制,Xue 等人[14]提出了隐私保护决策树分类(PDTC)方案,利用同态加密和秘密分享技术设计了一个安全比较协议.然而,由于大整数计算和多轮额外通信,PDTC 的效率不理想.此外,它需要一个可信的第三方,这是一个不切实际的假设.Zheng 等人[15]提出了另一种基于秘密分享的框架,改进了效率,并在预测过程中不需要提供者和客户之间的交互.作者进一步通过重新构造方法对其进行了优化,并将线性通信复杂度降低到对数复杂度[16].然而,以上所有框架都采用了同态加密或秘密分享技术,并未实现高效性.在本文中,我们提出了一种专注于安全数值比较而没有任何冗余操作(如加法或乘法)的方法.

3　背景知识

3.1　面基于决策树的模型

决策树是一种经典的非参数化监督机器学习算法,它由许多叶子节点表示预测结果和许多中间节点用于决策.其训练原则使用贪心算法在当前数据集中寻找最优的分裂策略,以最大化子分支的特定优化指标(例如信息增益[37]和基尼系数[38].我们以经典的二叉决策树 CART[39]为例阐述其预测过程,我们用 T 表示决策树模型,用 T_i 表示其第 i 个节点.如果 T_i 是一个叶子节点,则使用 $T_i{}^r$ 表示为预测的标签结果;否则,使用 $T_i{}^a$ 和 $T_i{}^v$ 表示该节点存储的分裂特征和阈值.使用 x 表示一个数据样本,将其第 i 个特征的值记为 $x[i]$.决策树的预测过程是自顶向下的模式:数据样本从根节点开始,在每个节点,决策树算法将当前节点的阈值与样本在对应特征的值进行比较,并根据比较结果选择不同的分支前进.例如,当 x 到达一个非叶子节点 T_i 时,x 将比较 $x[T_i{}^a]$与节点的存储阈值 $T_i{}^v$,以确定下一步的路线.这个过程重复进行,直到样本到达叶子节点,得到预测结果.上述将一个样本从根节点不断下落到叶子节点的流程在算法 1 进行了详细展示.

算法 1.　决策树预测流程

输入:决策树模型 T.
数据样本 x.
输出:预测结果.

```
1 BEGIN
2     m←T.root;
3     WHILE m 不是叶子节点 DO
4       IF x[m^a]<m^v THEN
5             m←m.left_child;
6       ELSE
7             m←m.right_child;
8       ENDIF
9     ENDWHILE
10     return m^r;
11 END
```

基于树的集成模型包括随机森林和梯度提升树(Gradient Boosting Decision Tree, GBDT).这两种模型都由多个普通决策树组成.不同之处在于随机森林是基于多样的特征子集训练一组决策树,而 GBDT 的子树则具有顺序关系,其中排名靠后的树的输出为其与前一棵树的残差.GBDT 有许多种流行的实现,例如 XGBoost[40],LightGBM[41]和 CatBoost[42].增量决策树是另一种专为增量场景设计的基于树的模型.它通常利用一些数学不等式(例如霍夫丁不等式)来实现对决策树的部分拟合.显著的增量决策树实现包括超快决策树(Very Fast Decision Tree, VFDT)[43]和极快决策树(Extremely Fast Decision Tree, EFDT)[44].以上所有基于树的模型的正确性都基于底层普通决策树是否能够正确训练.

3.2 揭序加密

与保序加密(Order-preserving Encryption, OPE)[45]不同,揭序加密[46]是一种不在密文中直接保留明文大小关系的加密算法.取而代之地,ORE利用一个公共比较函数揭示明文之间的顺序关系.具体而言,一套ORE方案包括以下组件:

ORE.INIT $\rightarrow sk$:初始化函数过程,生成用于后续加密的秘密密钥 sk .

ORE.ENC(x, sk) $\rightarrow \bar{x}$:加密算法,接受明文 x 和秘密密钥 sk 作为输入,并输出明文 x 的加密 $\bar{x}$.

ORE.CMP($\bar{x}$,$\bar{y}$) $\rightarrow r$:公开的比较函数,其中 $r \in \{-1,0,1\}$ 表示明文 x 和 y 之间的大小关系.

Chenette等人[46]基于伪随机函数提出了一种高效的ORE方案,称为CLWW.之后的ORE方案[47-48]大都是基于CLWW的思想进行改进以提高安全性和效率.CLWW方案已被研究者指出在数据库领域不够安全[49],然而,由于在本文中的场景中不需要数据库领域的区间查询功能且不要求密文可以解密,所提出的OREC可直接基于CLWW方案进行设计.

3.3 密钥协商

在密码学中,密钥协商是为了帮助两个实体在不安全的信道下建立安全通信的技术.通常,其包含密钥交换和密钥派生两个步骤,其中密钥交换帮助两个实体获得一个共享的秘密,密钥派生帮助其将生成的秘密扩展成适用于其他密码算法的长度.

3.3.1 密钥交换

密钥交换是一种用于在不安全的通信渠道上建立加密密钥交换的方法.我们使用Diffie-Hellman[50]协议(之后简称为 DH)来进行密钥交换.它基于由 g 生成的有限循环群 G,假设有两个参与方Alice和Bob,其工作流程可描述如下:

Alice和Bob各自选择一个随机值,即 a 和 b.然后,他们分别计算 g^a 和 g^b.

Alice将 g^a 发送给Bob,Bob也做同样的操作.

他们可以计算 $K_{ab}=(g^a)^b=(g^b)^a=g^{ab}$.

然后 K_{ab} 由Alice和Bob私下获得.我们用 $DH(Alice,Bob)$ 表示两个参与方Alice和Bob的密钥交换过程.

3.3.2 密钥派生

由于长度或其他因素,由 DH 生成的密钥通常不适合直接用于其它加密算法.密钥派生函数[51]旨在安全地生成更长的随机字节.在这项工作中,我们依赖一种这样的函数 KDF,它接受两个值 s_0 和 s_1 作为输入,并输出 $\bar{s}=KDF(s_0,s_1)$,其中 $\bar{s}$ 是一个具有足够比特位的大整数.

4 基于揭序加密的安全比较协议

在本节中,我们首先规范了联邦预测问题,并对其进行了符号表示.然后,我们介绍了问题模型,并在之后详细介绍了所提出OREC协议的细节.

4.1 联邦预测

在本小节中,我们以符号形式描述了联邦预测问题.该场景具有两个特点:

(1)数据异构性:假设存在一个模型提供方,拥有其私有模型 M;另有一个数据集 D,以及一组分享该数据集的数据拥有者集合 $D'=\{DO_i\}$.每个样本 $x \in D$ 都有若干特征,这些特征形成了特征集合 F.特征分布在所有数据所有者之间,这些数据所有者不允许与他人共享其数据.对于一个数据所有者 DO_i,我们其拥有的特征集合表示为 $F'(DO_i)$,其满足以下条件:

$$\bigcup_{i=1}^{|D'|} F'(DO_i)=F \tag{1}$$

以及

$$\forall i,j,F'(DO_i) \cap F'(DO_j)=\emptyset \tag{2}$$

(2)场景的动态特性使得上述数据集 D 可以不断增长,即数据所有者不断积累新的数据.

所描述的联邦预测问题是关于如何在上述场景且不泄露隐私的情况下,在任意时刻能够获得 $y_i=M(x_i)$, $x_i \in D'$ 的问题.在本工作中,我们将 M 视为基于树的模型.

4.2 问题场景

考虑有两个参与方Alice和Bob,他们分别拥有秘密值 a 和 b.他们希望在不将这些值相互透露的情况下进行大小比较.为了实现这一目标,他们将使用一个第三方的隐私计算平台 P,该平台提供协同计算服务.我们的目标是设计一个安全的比较协议 S,使得Alice和Bob能够获取 a 和 b 的比较结果,同时不将任何信息透露给彼此或 P.形式上,我们的目标是设计一个方案 S,其工作原理如下:

$$cmp(a,b)=S(P,Alice,a,Bob,b) \tag{3}$$

其中 $cmp(x,y) \in \{-1,0,1\}$ 表示两个值的比较结果.-1 表示 x 小于 y,1表示反之,而0表示 x 等于 y.

4.2.1 威胁模型

假设参与方Alice和Bob是半诚实(Semi-Honest)的,这意味着他们会老实地遵循协议执行,但可能尝试窃取他人的隐私信息.第三方计算平台 P 则

被视为恶意的,即它可能使用所有允许的技术来探查 Alice 和 Bob 的隐私,包括在协议中发送误导性的错误值等.

目前,对于保序加密和揭序加密,其理想的语义安全目标是有序情况下的对选择明文攻击的不可区分性(Indistinguishability Under an Ordered Chosen Plaintext Attack, IND-OCPA)[45]. 然而,不完备的揭序加密实现方式可能会泄漏如 MSB 之类的敏感信息,而攻击者可以利用这些信息来推导出更多的私密信息[49,52-53]. 因此,我们的目标是实现比 IND-OCPA 更强的隐私保证. 此外,我们假设提议的协议中的通信受到安全通信协议的保护,例如所有网络传输都已被传输层安全协议保护[54].

4.3　协议概览

所提出的算法的概述如图 2 所示,OREC 协议包含五个步骤,下面是每个步骤的简要描述:

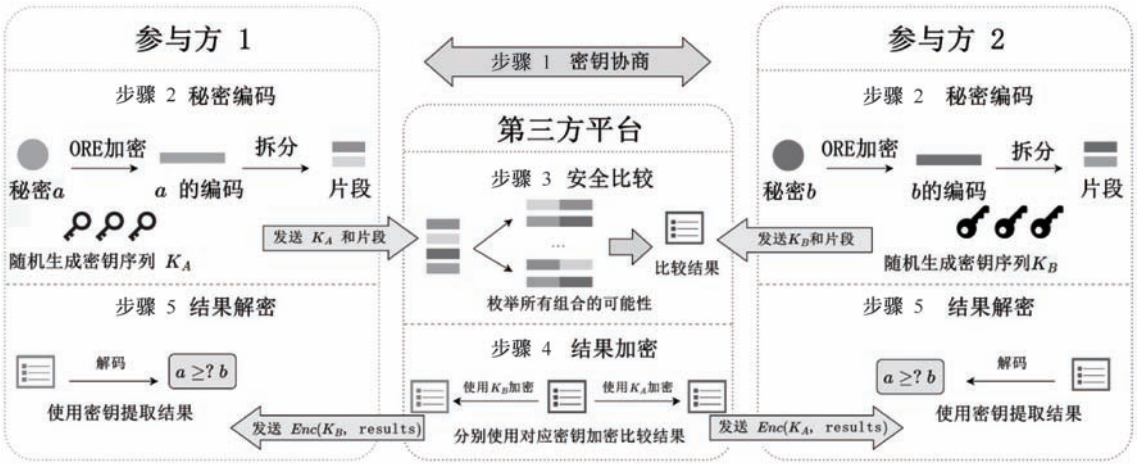

图 2　OREC 流程概述

(1)密钥协商:两个参与方安全地协商一个密钥,用于初始化一个伪随机数生成器(Pseudo-random Number Generator, PRNG),进而使用 PRNG 生成后续的共享秘密值.

(2)秘密编码:每个参与方使用选定的 ORE 算法对其待比较秘密值进行编码,并将结果拆分成几个部分. 然后,他们生成一个密钥序列,并将密钥和拆分后的片段一同发送给第三方.

(3)安全比较:第三方将接收到的片段,通过将它们连接成合法的 ORE 编码并枚举所有可能的方式来进行比较. 第三方最终会获得一个比较结果的序列.

(4)结果加密:第三方使用两个参与方生成的密钥列表,对比较结果按顺序进行加密,然后将两份加密结果发送给相应的参与方.

(5)结果解密:每个参与方可以使用协商的秘密位置对列表中的正确位置进行解密,最终得到真实的比较结果.

在接下来的子节中,我们将详细解释每个步骤的具体流程.

4.4　安全比较步骤

4.4.1　密钥协商

设 KDF 为一个公开的密钥派生函数. 在这一步中,作为参与方的 Alice 和 Bob 进行了后续任务所需的预处理流程. 他们首先协商出一个共享的私密密钥 k_0,该密钥对第三方 P 保密. 接下来,Alice 和 Bob 派生出一个种子 $sd = KDF(k_0)$. 随后,他们得以共享一个以 sd 为初始值 PRNG. 我们用 G 表示该共享的 PRNG,用 $G(a)$ 表示利用其生成一个在[0, a)范围内的整数,其中 a 是正整数. 利用 G,Alice 和 Bob 生成以下所需的共享秘密值:

ORE 加密密钥 $k_1 = G(2^{64})$:需要注意的是,对于更高的安全级别,2^{64} 可以替换为更大的数值.

秘密位置 $p = G(4)$:p 的值确定了分割片段的排列方式,稍后将会解释.

两个验证值 a' 和 b':它们都使用 $G(2^{64})$ 生成. 这些值用于检测 P 是否存在破坏行为.

4.4.2　秘密编码

我们采用 CLWW[46] 提出的按位编码方法作为底层的 ORE 算法. 在之后的描述中,我们将 CLWW 中的编码和比较过程分别被表示为 ORE.ENC 和 ORE.CMP. CLWW 的核心思想是使用数位的位置信息加密数字的每一位,并用加密的模运算结果替换原始位. 由于每个原始位要么为 0,要么为 1,比较过程即找到两个编码第一个不一样的四进制位,并

通过在一个数下比较哪个数字比另一个数字大 1 来判断谁更大. 我们在算法 2 和 3 中给出了 CLWW 算法的伪代码.

算法 2. CLWW 编码算法

输入:哈希函数 H;
加密密钥 k;
待加密值 x.
输出:x 的密文全 $\hat{x}$.

```
1 BEGIN
2     x_1 x_2 ⋯ x_n ← x 的二进制表示;
3     FOR i∈[1,n] DO
4         u_i ←
            H(k⊕i⊕x_1 x_2 ⋯ x_{i-1}⊕0^{n-i})+x_i;
            /* ⊕ 表示字符串的链接, 0^m 表示 0 重复
            m 遍.          */
5         u_i ← u_i  mod 4;
6     ENDFOR
7     x ← u_1 u_2 ⋯ u_n;
8     return x̂;
9 END
```

以 Alice 作为示例来描述秘密编码过程. 首先,Alice 对她的秘密值 a 进行加密,并获得 a^*,其中 $a^* = ORE.ENC(k_1, a)$. 加密后的值 a^* 可以用四进制格式表示,如下所示:

$$a^* = a^*_{n-1} a^*_{n-2} \cdots a^*_1 a^*_0 \tag{4}$$

其中,n 是 a 的四进制长度,每个 $a_i^* \in [0, 3]$.

然后,Alice 将 a^* 分为两部分,分别表示为 a_l^* 和 a_r^*,如下所示:

$$a_l^* = a^*_{n-1} a^*_{n-2} \cdots a^*_{\lfloor \frac{n}{2} \rfloor} \tag{5}$$

$$a_r^* = a^*_{\lfloor \frac{n}{2} \rfloor - 1} a^*_{\lfloor \frac{n}{2} \rfloor - 2} \cdots a^*_0 \tag{6}$$

然后,Alice 使用 G 连续生成两个密钥 k_a 和 k_b. k_a 和 k_b 分别称为 Bob 和 Alice 的解码密钥. 此外,Alice 使用自己私密的种子生成额外三个随机密钥,分别记为 k_{a0}、k_{a1} 和 k_{a2}. 这三个密钥称为干扰密钥. 干扰密钥的引入使得 Bob 无法确认哪一个密钥对 Alice 是真实有效的,从而防止了其通过共享的解密密钥还原出 Alice 的其余比较结果. 该信息的泄露使得 Bob 能通过一些攻击手段获取更多 Alice 隐私数据的有效位. 干扰密钥的引入使得 Bob 无法确认哪一个密钥对 Alice 是真实有效的,从而防止了其通过共享的解密密钥还原出 Alice 的其余比较结果. 该信息的泄露使得 Bob 能通过一些攻击手段获取更多 Alice 隐私数据的有效位. 解码密钥 K_a 和三个干扰密钥构成了 Alice 的密钥序列 k_A,Alice 保证 K_a 的第 p 个元素一定为 k_a,其中 p 是在密钥协商步骤中生成的共享秘密位置. K_a 中的其他位置随机由扰动密钥占据. 最后,Alice 将元组 (a_i^*, a_j^*) 和 K_a 发送给 P,其中 $i, j \in \{0, 1\}$,且 $i \neq j$. a_l^* 和 a_r^* 的顺序由以下规则确定:如果 $p \leqslant 1$ \$,则 $i = 0$ 且 $j = 1$. 否则,$i = 1$ 且 $j = 0$. 请注意,此规则不适用于 Bob.

对于 Bob,其大部分工作流程与 Alice 相同. 其中的差异如下:首先,Bob 的解码密钥是 k_b,而不是 k_a. 其次,元组 (b_i^*, b_j^*) 的顺序规则不同:如果 $p \in \{0, 2\}$,则 $i = 0$ 且 $j = 1$. 否则,$i = 1$ 且 $j = 0$. 这些条件在表 1 中进行了说明. 在该样例中,我们为只有 2 个片段的情况提供了示例. 而实际中,所提出的协议也支持更多片段的情况. 在这种情况下,这些片段的元组将按片段索引的字典序进行排序.

表 1 分割段数为 2 时 ORE 编码片段的顺序规则

条件	Alice	Bob
$p = 0$	(a_l^*, a_r^*)	(b_l^*, b_r^*)
$p = 1$	(a_l^*, a_r^*)	(b_r^*, b_l^*)
$p = 2$	(a_r^*, a_l^*)	(b_l^*, b_r^*)
$p = 3$	(a_r^*, a_l^*)	(b_r^*, b_l^*)

算法 3. CLWW 比较算法

输入:两个待比较密文 $\hat{x}$ 和 $\hat{y}$.
输出:比较结果.

```
1 BEGIN
2     IF x̂ = ŷ THEN
3         return 0;
4     ENDIF
5     x_1 x_2 ⋯ x_n ← x̂ 的四进制表示;
6     y_1 y_2 ⋯ y_n ← ŷ 的四进制表示;
7     i ← 最小满足 x_i ≠ y_i 的位;
8     IF x_i = (y_i + 1) mod 3 THEN
9         return 1;
10    ELSE
11        return − 1;
12    ENDIF
13 END
```

4.4.3 安全比较

当第三方 P 收到来自 Alice 和 Bob 的 ORE 编码片段后,其会启动比较过程. 首先,它会枚举两个参与者的所有可能连接方式,该步骤通过枚举每种可能的片段顺序来实现. 例如,如果 P 从某个参与者收到两个片段,则会尝试 $2! = 2$ 种情况. 每个连接方式都被视为可能的原始值 ORE 编码. 此枚举按照字典顺序进行,例如对于 n 个片段,计数时首

先考虑的连接方式是(1, 2,…, n),最后则为 (n, n −1, …,1).

设 E_A 和 E_B 分别表示 Alice 和 Bob 的连接集合.对于 E_A 中的每个元素 u 和 E_B 中的每个元素 v,P 将调用 $ORE.CMP$ 函数生成一个比较结果.比较结果的集合记为 R,表示为

$$R=\{ORE.CMP(u,v)\mid \forall u\in E_A, v\in E_B\} \tag{7}$$

4.4.4　结果加密

P 分别使用 K_A 和 K_B 将比较结果集 R 进行两次加密.例如,在第一次中,R 中的第 i 个结果 r_i 会用 K_A 中的第 i 个密钥进行加密,最终得到加密的结果列表 R^A.类似地,R^B 是使用 K_B 对 R 进行加密得到的.最后,P 将 R^A 发送给 Bob,将 R^B 发送给 Alice.

4.4.5　结果解密

Alice 和 Bob 可以使用以下方法推导出 a 和 b 的真实比较结果:

$$ORE.CMP(a^*, b^*)=\mathrm{DEC}(K_B, R_p^A) \tag{8}$$

$$=\mathrm{DEC}(K_A, R_p^B) \tag{9}$$

其中 p 是 P 用于比较的连接元组的索引.

如前一部分所述,连接的枚举将按索引的字典序进行,并且第 p 个枚举恰好可以被还原出正确的原始值 ORE 编码.因此,K_A 和 K_B 中的第 p 个密钥是 Alice 和 Bob 之间的共享秘密.通过使用相应的密钥对 R^A 和 R^B 中的第 p 个元素进行解密,Alice 和 Bob 均可以获得正确的比较结果.

4.4.6　结果验证

本小节介绍了如何使用验证值 a' 和 b' 来防止 P 提供错误的答案.在“密钥协商”步骤中,Alice 和 Bob 分别对 a' 和 b' 进行相同的编码过程.因此,除了发送目标值和密钥列表的编码片段元组之外,他们还将 a' 和 b' 的编码片段元组发送给 P.在“安全比较”步骤中,P 会基于接收到的另一份片段元组计算额外的结果比较结果,并以加密形式发送给 Alice 和 Bob.由于 a' 和 b' 是 Alice 和 Bob 之间的共享秘密,而 P 无法区分接收到的元组哪个为真实值,哪个为验证值,因此 Alice 和 Bob 可以检查 a' 和 b' 的比较结果,以确认 P 没有提供错误的答案.

算法 4. OREC 的批处理模式

输入:N_A 和 N_B,即 Alice 和 Bob 的待比较值序列;
批处理大小 b.
输出:比较结果序列 R.

```
1  BEGIN
2    Alice 和 Bob 将 N_A 和 N_B 分为若干批次;
     /*  将 N_A 和 N_B 的笛卡尔积表示
     为(N_A, N_B).                    */
3    FOR Batch_i ∈ (N_A, N_B) DO
4        Alice 和 Bob 协商出 k_0;
5        Alice.ORE.ENC(k_0, Batch_i.A);
6        Bob.ORE.ENC(k_0, Batch_i.B);
7        Alice 和 Bob 实施 OREC;
8        R_i ← 比较结果;
9    ENDFOR
10   R ← ∪R_i;
11   return R;
12 END
```

4.5　批处理模式

在实际场景中,当 Alice 和 Bob 需要比较大量的数值对时,执行多个密钥协商步骤将拉低效率.为解决这个问题,OREC 支持批处理模式,可以在一轮中比较多个数值对.在批处理模式中,批处理大小用 b 表示,表示一轮中要比较的秘密值对的数量.工作流程如算法 4 所示.在每一轮中,Alice 和 Bob 协商一个新的 k_0 来初始化一个新的 PRG.然后使用该密钥对批处理中的秘密值进行加密.通过利用批处理模式,Alice 和 Bob 的通信需求大大减少.通过选择合理的 b 值,OREC 的隐私仍然可以抵御统计攻击[49].这种模式可以提高在需要执行大量比较的场景中的效率.若不采用批处理模式,则每对私密值的比较均需要通过密钥协商,即每 b 次比较相对于批处理模式多了 $b-1$ 次通信.

4.6　隐私分析

4.6.1　对抗不可信任的第三方

首先,我们分析 P 可以从接收到的中间值中预测出什么信息.P 从每个参与者那里接收到两对 ORE 编码片段,一对是用于秘密值 a,另一对是用于验证值 a'.由于 P 无法区分哪一对是用于秘密值的,其猜测的期望成功率为 $\frac{1}{2}$.对于每一对编码片段,攻击者 P 仍然无法确认正确的连接方式,其猜测的成功率仍为 $\frac{1}{2}$.因此,P 预测出用于比较的真实编码对的成功率为 $(\frac{1}{2})^4=\frac{1}{16}$.在其它情况下,如果参与者选择将 ORE 编码分成 k 个片段,猜测的成功率将降至 $\frac{1}{4(k!)^2}$. 此外,如果比较操作重复进行 N 次,OREC 的隐私级别下降到仅与 ORE 相

同的预期速率估计为 $\frac{1}{4(k!)^{2N}}$.

总之,当片段数设置为 2 时,一轮 OREC 下降到 ORE 的安全级别(即 IND-OCPA)的概率为 $\frac{1}{16}$. 由于该方案是一次一密的形式,P 无法像先前的工作中收集多个密文-明文对那样进行攻击[49,52]. 因此,OREC 可以抵抗不可信任的第三方的隐私攻击.

4.6.2 对抗半诚实参与者

我们设计的干扰密钥序列方案防止了参与者获取除最终比较结果以外的任何额外信息. 这是因为除了真实结果之外的所有比较结果都是使用随机生成的密钥加密的,因此,参与者无法从除真实比较结果之外的其它比较结果中预测出另一方的其它信息.

4.7 效率分析

首先,我们分析 OREC 的计算效率. 假设秘密值和验证值的长度为 n 比特,则一轮的 ORE 编码的时间复杂度为 $O(n)$. 如果我们使用位异或来加密和解密比较结果,那么它们的复杂度也是 $O(n)$. 因此,总体计算复杂度为 $O(n)$. 具体而言,每个参与者执行两次 ORE 加密和两次异或解密. 在安全比较步骤中,P 对秘密值和验证值进行了 $2(k!)^2$ 次 ORE 比较过程的枚举,其中 k 表示 ORE 编码元组中的片段数量. 然后,它进行了 $2(k!)^2$ 次 XOR 加密操作. 因此,第三方的计算复杂度为 $O((k!)^2n)$,其中 k 通常不大于 3.

接下来,我们分析 OREC 的通信开销. 密钥协商步骤需要 Alice 和 Bob 交换一个秘密密钥 k_0. 假设他们使用 Diffie-Hellman[50] 方案,则参与者的通信负载为 $O(N)$,其中 N 是 DH 中选择的大整数的位长度. 由于 $ORE.CMP$ 的结果长度为 3 bit,基于异或方案的扰动密钥和编码/解码密钥也为 3 bit. 因此,参与者除了发送 $2\times n\times 2$ bit 的编码片段外,还会发送 $2\times(k!)^2\times 3$ bit 的密钥. 因此,参与者的通信开销为 $6\times(k!)^2+4\times n$ bit,即 $O((k!)^2+n)$. 对于第三方,它只需发送 $6\times(k!)^2$ bit 的加密比较结果,其通信开销为 $O((k!)^2)$. 在分析了通信负载后,我们进一步分析通信次数. 每个参与者在密钥协商步骤中需要发送一次消息,并在秘密编码步骤中再进行一轮通信. 因此,每个 OREC 执行中,参与者需要进行 2 轮通信. 第三方也需要进行 2 次消息交换,分别在 Alice 和 Bob 的结果编码步骤中. 复杂度分析如表 2 所示. 总而言之,所提出的 OREC 在计算和通信复杂度上都具有极低的开销.

表 2 OREC 的时间复杂度

	参与方	第三方
计算复杂度	$O(n)$	$O((k!)^2n)$
通信复杂度	$O(N+(k!)^2+n)$	$O((k!)^2)$

5 实验分析

5.1 环境设置

5.1.1 硬件

所提出的算法在一台运行 Linux 的普通计算机上实现和评估,该计算机配备了 Intel(R) Xeon(R) Gold 6248R CPU 和 160 GB 内存. 实验在局域网环境中进行,以确保高效的网络通信.

5.1.2 底层算法与参数

在 ORE 方案中,我们使用 Blake2[55] 哈希函数进行加密操作,并在密钥协商步骤中使用 Argon2[56] 密钥派生函数. 这些经过充分验证的加密组件为 OREC 算法提供了强大的安全保证. 关于表示 ORE 编码片段数量的参数 k,除非另有说明,我们在实现中将其设置为 2. 这个选择在计算效率和安全性考虑之间取得了平衡.

5.2 场景设置

在两个不同的场景中对所提出的方法进行评估,以评估其性能和实用性. 第一个场景是安全比较场景,旨在评估 OREC 算法的正确性和效率. 该场景设置为批量的“百万富翁问题”,即多组纯粹的数值大小比较任务. 第二个场景是外包决策树预测场景,旨在评估 OREC 的实用性. 该场景即面向联邦学习的延伸场景联邦预测任务. 在该任务中,我们假设存在数据拥有方、模型提供方和隐私计算平台三个实体. 为了评估所提出方法的泛用性,我们在大量流行的决策树模型和多种数据集上进行了测试.

5.2.1 模型

在外包决策树预测场景的评估中,我们考虑了几种常用的模型,包括普通决策树、随机森林、XGBoost[40] 和 CatBoost[42]. 这些模型的实现基于 scikit-learn 库[57],底层的决策树采用 CART[39] 算法.

5.2.2 数据集

本章节中我们仍使用 UCI 机器学习库[58] 和 LIBSVM[59] 中的若干数据集来验证所提出 OREC 协议的鲁棒性和泛用性. 由于安全比较协议仅支持

整型数据,因此需要将数据集中的浮点数处理为整数.处理方法为将浮点数乘以一个大数后截断,在本章节中,所选取的乘数为 2^{16}.这些数据集的基本信息总结如表 3 所示,其中 Iris 和 Cod-RNA 数据集中的浮点数已被处理成整数,且 Adult 数据集中的字符串类型数据通过 one-hot[60] 算法处理成了整数.这些数据集中的值主要包括整数或浮点数,涵盖了多种数值范围,便于我们全面地测试所提出协议的正确性与泛用性.

表 3　OREC 测试使用的数据集

	样本数量	特征数量	数据类型
Iris	150	4	Float
Madelon	2,600	500	Integer
Adult	48,842	14	Integer
Cod-Rna	59,535	8	Float

5.3　基　线

在安全比较场景中,我们将我们提出的 OREC 方法与 SecureNN[32] 进行比较,SecureNN 是目前最先进的安全多方计算框架之一. SecureNN 和 OREC 一样,涉及一个第三方的参与者,但与 OREC 不同的是,在 SecureNN 中,所有参与者具有相同身份,且其可以防止合谋攻击.

在外包决策树预测场景中,将我们提出的方法 OREC 与 Zheng 等人[16] 提出的解决方案进行测试,其为目前在安全决策树预测场景中的最先进方法之一.我们将 Zheng 等人的方法称为 OSDTI(Optimized Secure Decision Tree Inference).类似于 OREC,OSDTI 也利用外包云服务器进行协助比较.OSDTI 的基本原理是确定要比较的两个值之间的最高有效位,其比较方法采用了 De 等人[61] 的思想,并设计了一个附加的过程,用于将秘密共享的份额从一个域转换到另一个域.为了方便比较和评估,我们使用 Python 编程语言实现了 OSDTI 的比较阶段与 OREC 进行对比.

5.4　安全比较

使用不同长度的数值对 OREC 进行了性能评估.具体而言,我们分别选择了 16、32 和 64 位的值,这些长度的值代表了常用的数据类型.实验同时对整数和浮点数进行了评估,其中浮点数经过了乘以一个比例因子的预处理.

实验结果如图 3(a)所示,显而易见,协议执行的时间开销与数值的比特长度和执行轮数均呈线性增长.就平均情况而言,OREC 比较 1000 对秘密值的时间花费不到 2 秒,这体现了 OREC 极高效率.如之前所述,参数 k 会显著影响时间复杂度,因为其对复杂度的贡献为 $(k!)^2$.因此,我们使用不同 k 值进行了多组测试,结果如图 3(b)所示,该批次实验使用了 32 比特的值,其实验结果清楚地表明,随着 k 值的增大,时间开销显著增加.因此,在 OREC 中将 k 设置为不大于 3 是可以接受的.

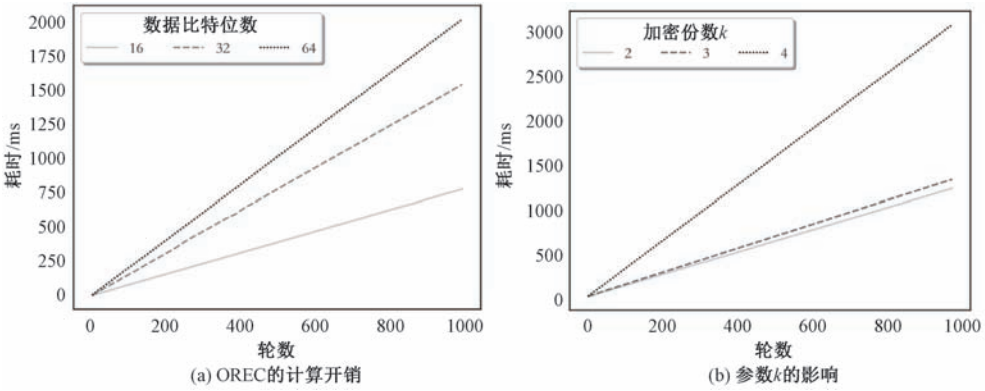

图 3　计算开销与不同的数据类型和 k 的关系

此外,我们还将 OREC 与 SecureNN 框架的比较模块进行了比较,该框架通常用于隐私神经网络训练[32].我们使用 Python 实现了 SecureNN 的安全比较算法,以便与所提出的 OREC 进行计算时间的对比实验.结果如图 4 所示,观察可知,当数据规模较小时,SecureNN 的效率与 OREC 相似;但由于 SecureNN 的时间复杂度高度依赖于比特长度,导致了当数据规模增大时其时间开销显著增加.例如,在 64 位的数据上,SecureNN 的时间成本约为 OREC 的 3.4 倍.在科学计算的诸多领域中,常常会遇到需要处理更大规模数据的时候,例如需要对 128、256,甚至更高位的数值作处理.因此,OREC 相较于 SecureNN 不仅在计算效率上具有显著的优势,也具有更宽阔的应用场景.此外,上述实验仅对

比了计算耗时，而未考虑通信时间成本. 由于 SecureNN 相较于 OREC 需要大量的通信轮次并具有更高的负载规模，OREC 在通信和计算效率方面均优于 SecureNN.

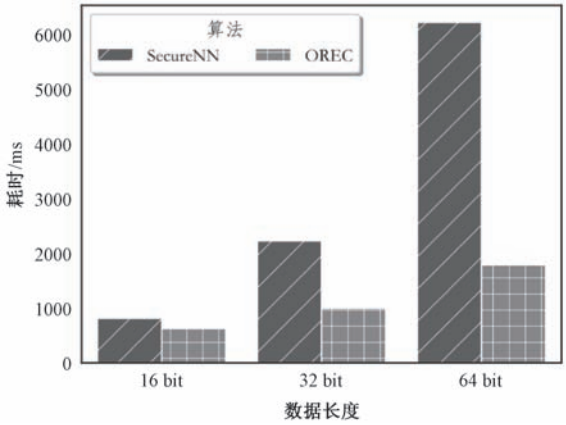

图 4 SecureNN[32] 与 OREC 的效率对比
（用于处理 1000 个比特的时间消耗）

5.5 外包决策树预测

为方便起见，在接下来的实验中我们将特定的模型与相应的数据集进行了配对绑定. 具体而言，我们使用原始的决策树模型在 Iris 数据集上进行测试，使用随机森林模型在 Madelon 数据集上进行测试，使用 XGBoost 模型在 Adult 数据集上进行测试，以及使用 CatBoost 模型在 Cod-RNA 数据集上进行测试. 在后续讨论中，若无特殊说明，这些模型—数据集的配对将保持不变.

首先，我们进行了实验以证明 OREC 的正确性，结果如表 4 所示. 实验结果表明了所提出的协议对所选数据集的预测任务均无造成任何精度损失. 此外，我们也具体追踪了 OREC 和原始模型中的数据在树模型上的传递过程，并确认了我们的方法与原始模型对数据的处理流程完全一致.

表 4 OREC 与原始模型准确度比较（单位：%）

数据集	原始	OREC
Iris	94.17	94.16
Madelon	70.67	70.67
Adult	83.85	83.85
Cod-RNA	96.63	96.63

接下来，我们将 OREC 与最先进的外包决策树预测框架之一 OSDTI 进行了比较. 在该场景，我们将 Alice 视为决策树模型提供者，Bob 视为数据所有者. 其中 Bob 希望使用 Alice 的模型对自己的数据进行预测，他们借助一个不可信的第三方来完成这一目的. 因为 Iris 数据集对应的决策树模型只有一颗树并且其训练模型的深度较小，我们在实验中对 Iris 数据集进行了 10000 次数据预测，而对其他数据集进行了 100 次数据预测. 其中除原始决策树模型以外的集成模型中的子树数量被设置为了 100，并且所有模型在训练时均未设置深度限制. 实验结果显示在图 5 中. 两种方法都证明了它们对精度是无损的. 关于时间消耗的发现表明，所提出的方法在效率上对比 OSDTI 具有极大优势，因为它需要很少的通信次数. 通过计算与通信比例也可以看出，OSDTI 中的额外通信开销是导致其低效的主要因素.

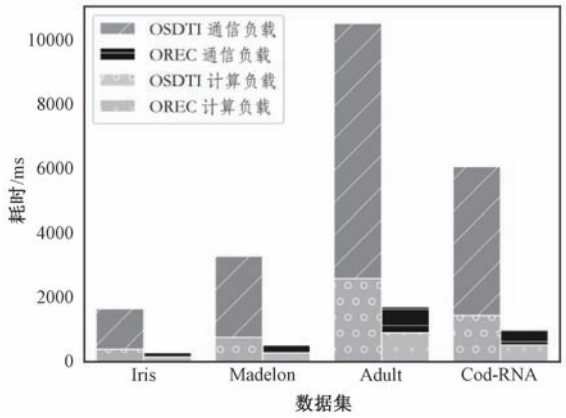

图 5 OSDTI 和 OREC 的效率比较

总而言之，所提出的 OREC 框架在外包决策树预测任务中于通信和计算效率方面都具有明显的优势，超过了目前的其他算法.

6 讨 论

6.1 解耦组件

在本文中，我们通过利用揭序加密的特性设计了实用的安全比较协议 OREC. 出于对轻量级的考虑，我们选择了 CLWW[46] 作为底层的揭序加密方案. 然而值得注意的是，许多研究者已对 CLWW 方案已经了深入研究，并证明了其在外包存储的应用场景未能达到令人满意的安全性[49,53,62]. 在 OREC 中，尽管一次一密机制和结果混淆技术的结合大大缓解了这些困境，但在极端情况下，OREC 的安全级别仍可能降低到与 CLWW 同等的安全级别. 因此，在实际应用中，一些研究人员可能会倾向于使用更新的、具有更高安全性的其他揭序加密算法. 由于所提出的 OREC 框架具有高度解耦的特性，解决该问题十分简单. 在 OREC 包含的诸多组件中，底层揭序加密算法可以随意地被替换为其他揭序加密方

案,例如 EncodeORE[48]等,以增强整体安全性并满足特定应用的要求.

6.2 应用场景

与传统的基于安全多方计算的解决方案相比,我们提出的 OREC 协议通过引入了不可信第三方的参与,实现了更高的效率.显而易见,这种设计引入了在执行效率和对合谋攻击的抵抗性之间的一种权衡.然而,在绝大多数当代的隐私计算服务中,由于参与计算的实体之间难以具有充足的计算力和顺畅的通信环境,其与陌生的第三方平台产生交互的情况是不可避免的.因此,研究如何组织各方之间进行可靠的合作对于开发和部署私密计算平台至关重要.

7 结　论

外包隐私计算和联邦学习等技术的进步极大地推动了关于在保护隐私的前提下最大化数据利用率的研究.本文中我们提出了 OREC,一个新颖的安全比较协议.通过精心设计的通信策略与揭序加密方案的有效利用,OREC 在拥有较高安全性的前提下具有极高的运行和通信效率.OREC 可应用于多种场景,包括隐私保护机器学习和外包模型预测服务等.通过进行大量实验,我们证明了 OREC 的有效性和正确性,并突显了其作为隐私计算平台中强大部件的潜力.我们坚信隐私计算技术在信息时代中扮演着重要角色,进一步在这个领域进行研究具有重要的价值.

参　考　文　献

[1] Liang J, Qin Z, Xiao S, et al. Efficient and secure decision tree classification for cloud-assisted online diagnosis services. IEEE Transactions on Dependable and Secure Computing, 2021, 18(4): 1632-1644

[2] Yap B W, Ong S H, Husain N H M. Using data mining to improve assessment of credit worthiness via credit scoring models. Expert Systems with Applications, 2011, 38(10): 13274-13283

[3] Voigt P, Von dem bussche A. The eu general data protection regulation (gdpr). A Practical Guide, 1st Ed., Cham: Springer, 2017, 10: 3152676

[4] Pardau S L. The california consumer privacy act: Towards a european-style privacy regime in the united states. Journal of Technology Law and Policy, 2018, 23: 68

[5] Fu F, Shao Y, Yu L, et al. Vf^{2}boost: Very fast vertical federated gradient boosting for cross-enterprise learning//Proceedings of the International Conference on Management of Data. Virtual Event, China, 2021: 563-576

[6] Dai M, Zheng Z, Hong Z, et al. Edge computing aided coded vertical federated linear regression. IEEE Transactions on Cognitive Communications and Networking, 2022, 8(3): 1543-1551

[7] Dai M, Xu A, Huang Q, et al. Vertical federated dnn training. Physical Communication, 2021, 49: 101465

[8] Acar A, Aksu H, Uluagac A S, et al. A survey on homomorphicencryption schemes: Theory and implementation. ACM Computing Surveys, 2018, 51(4). DOI: 10.1145/3214 303

[9] Zhao C, Zhao S, Zhao M, et al. Secure multi-party computation: Theory, practice and applications. Information Sciences, 2019, 476: 357-372

[10] Cheng K, Fan T, Jin Y, et al. Secureboost: A lossless federated learning framework. IEEE Intelligent Systems, 2021, 36(6): 87-98

[11] Wu Y, Cai S, Xiao X, et al. Privacy preserving vertical federated learning for tree-based models. Proceedings of the VLDB Endowment, 2020, 13(11): 2: 2090-2103

[12] Mohassel P, Rindal P. Aby3: A mixed protocol framework for machine learning//Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. Toronto, Canada, 2018: 35-52

[13] Cock M D, Dowsley R, Horst C, et al. Efficient and private scoring of decision trees, support vector machines and logistic regression models based on pre-computation. IEEE Transactions on Dependable and Secure Computing, 2019, 16(2): 217-230

[14] Xue L, Liu D, Huang C, et al. Secure and privacy-preserving decision tree classification with lower complexity. Journal of Communications and Information Networks, 2020, 5(1): 16-25

[15] Zheng Y, Wang C, Wang R, et al. Optimizing secure decision tree inference outsourcing. IEEE Transactions on Dependable and Secure Computing, 2022,20(4): 3079-3092

[16] Zheng Y, Duan H, Wang C, et al. Securely and efficiently outsourcing decision tree inference. IEEE Transactions on Dependable and Secure Computing, 2022, 19(3): 1841-1855

[17] Kerschbaum F, Blass E, Mahdavi R A. Faster secure comparisons with offline phase for efficient private set intersection//Proceedings of the 30th Annual Network and Distributed System Security Symposium. San Diego, USA, 2023

[18] Al-rubaie M, Chang J M. Privacy-preserving machine learning: Threats and solutions. IEEE Security & Privacy, 2019, 17(2): 49-58

[19] Garay J A, Schoenmakers B, Villegas J. Practical and secure solutions for integer comparison//Public Key Cryptography-PKC 2007, 10th International Conference on Practice and Theory in Public-Key Cryptography. Beijing, China, 2007: 330-342

[20] Bellare M, Hoang V T, Rogaway P. Foundations of garbled circuits//Proceedings of the 2012 ACM Conference on Computer and Communications Security. Raleigh, USA, 2012: 784-796

[21] Beimel A. Secret-sharing schemes: A survey//Coding and Cryptology: Third International Workshop, IWCC 2011, Qingdao, China, 2011: 11-46

[22] Poh G S, Chin J J, Yau W C, et al. Searchable symmetric encryption: Designs and challenges. ACM Computing Surveys, 2017, 50(3): 1-37

[23] Xu L, Duan H, Zhou A, et al. Interpreting and mitigating leakage-abuse attacks in searchable symmetric encryption. IEEE Transactions on Information Forensics and Security, 2021, 16(3): 5310-5325

[24] Oya S, Kerschbaum F. IHOP: Improved statistical query recovery against searchable symmetric encryption through quadratic optimization//Proceedings of the 31st USENIX Security Symposium (USENIX Security 22). Boston, USA, 2022: 2407-2424

[25] Sun X, Yu F R, Zhang P, et al. A survey on zero-knowledge proof in blockchain. IEEE Network, 2021, 35(4): 198-205

[26] Catrina O, De hoogh S. Improved primitives for secure multiparty integer computation//Proceedings of the Security and Cryptography for Networks. Berlin, Germany, 2010: 182-199

[27] Kerschbaum F, Biswas D, De hoogh S. Performance comparison of secure comparison protocols//Proceedings of the 2009 20th International Workshop on Database and Expert Systems Application. Linz, Austria, 2009: 133-136

[28] Zahur S, Evans D. Obliv-c: A language for extensible data-oblivious computation. Cryptology ePrint Archive, 2015

[29] Hussain S, Li B, Koushanfar F, et al. Tinygarble2: Smart, efficient, and scalable yao's garble circuit//Proceedings of the 2020 Workshop on Privacy-Preserving Machine Learning in Practice. USA, 2020: 65-67

[30] Rabin M O. How to exchange secrets with oblivious transfer. Cryptology ePrint Archive, 2005

[31] Schoenmakers B. MPyC—python package for secure multiparty computation//Workshop on the Theory and Practice of MPC. 2018

[32] Wagh S, Gupta D, Chandran N. SecureNN: 3-party secure computation for neural network training. Proceedings on Privacy Enhancing Technologies, 2019, 2019(3): 26-49

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

[34] Veugen T, Blom F, De hoogh S J A, et al. Secure comparison protocols in the semi-honest model. IEEE Journal of Selected Topics in Signal Processing, 2015, 9(7): 1217-1228

[35] Paillier P. Public-key cryptosystems based on composite degree residuosity classes//Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques: Vol. 1592. Prague, Czech Republic, 1999: 223-238

[36] Cheon J H, Kim A, Kim M, et al. Homomorphic encryption for arithmetic of approximate numbers//Advances in Cryptology-ASIACRYPT 2017: 23rd International Conference on the Theory and Applications of Cryptology and Information Security. Hong Kong, China, 2017: 409-437

[37] Kent J T. Information gain and a general measure of correlation. Biometrika, 1983, 70 (1): 163-173

[38] Yuan Y, Wu L, Zhang X. Gini-impurity index analysis. IEEE Transactions on Information Forensics and Security, 2021, 16(3): 3154-3169

[39] Lewis R J. An introduction to classification and regression tree (cart) analysis//Proceedings of the Annual Meeting of the Society for Academic Emergency Medicine. San Francisco, USA, 2000

[40] Chen T, Guestrin C. Xgboost: A scalable tree boosting system//Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Francisco, USA, 2016: 785-794

[41] Ke G, Meng Q, Finley T, et al. Lightgbm: A highly efficient gradient boosting decision tree //Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017. Long Beach, USA, 2017: 3146-3154

[42] Prokhorenkova L O, Gusev G, Vorobev A, et al. Catboost: unbiased boosting with categorical features//Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018. Montreal, Canada, 2018: 6639-6649

[43] Domingos P M, Hulten G. Mining high-speed data streams//Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Boston, USA, 2000: 71-80

[44] Manapragada C, Webb G I, Salehi M. Extremely fast decision tree//Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Boston, USA, 2000: 71-80

[45] Boldyreva A, Chenette N, Lee Y, et al. Order-preserving symmetric encryption//Proceedings of the Advances in Cryptology-EUROCRYPT 2009. Berlin, Germany, 2009: 224-241

[46] Chenette N, Lewi K, Weis S A, et al. Practical order-revealing encryption with limited leakage//Fast Software Encryption: 23rd International Conference, FSE 2016. Bochum, Germany, 2016: 474-493

[47] Lewi K, Wu D J. Order-revealing encryption: New constructions, applications, and lower bounds//Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. New York, USA, 2016: 1167-1178

[48] Liu Z, Lv S, Li J, et al. Encodeore: Reducing leakage and preserving practicality in order-revealing encryption. IEEE Transactions on Dependable and Secure Computing, 2022, 19

(3): 1579-1591

[49] Wang X, Zhao Y. Order-revealing encryption: file-injection attack and forward security//Computer Security: 23rd European Symposium on Research in Computer Security, ESORICS 2018, Barcelona, Spain, 2018: 101-121

[50] Maurer U M, Wolf S. The diffie-hellman protocol. Designs, Codes and Cryptography, 2000, 19(2-3): 147-171

[51] Krawczyk H. Cryptographic extraction and key derivation: The hkdf scheme//Proceedings of the Annual Cryptology Conference. Santa Barbara, USA, 2010: 631-648

[52] Jurado M, Palamidessi C, Smith G. A formal information-theoretic leakage analysis of order-revealing encryption//Proceedings of the 2021 IEEE 34th Computer Security Foundations Symposium(CSF). Dubrovnik, Croatia, 2021: 1-16

[53] Grubbs P, Sekniqi K, Bindschaedler V, et al. Leakage-abuse attacks against order-revealing encryption//Proceedings of the 2017 IEEE symposium on security and privacy (SP). San Jose, USA, 2017: 655-672

[54] Dierks T, Rescorla E. RFC 5246: The transport layer security (tls) protocol version 1.2. USA: RFC Editor, 2008

[55] Aumasson J P, Neves S, Wilcox-o'hearn Z, et al. BLAKE2: Simpler, smaller, fast as md5//International Conference on Applied Cryptography and Network Security. Banff, Canada, 2013: 119-135

[56] Biryukov A, Dinu D, Khovratovich D. Argon2: new generation of memory-hard functions for password hashing and other applications//2016 IEEE European Symposium on Security and Privacy (EuroS&P). Saarbrücken, Germany, 2016: 292-302

[57] Pedregosa F, Varoquaux G, Gramfort A, et al. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 2011, 12: 2825-2830

[58] Dua D, Graff C. UCI machine learning repository. School of Information and Computer Sciences, University of California, Irvine, 2017

[59] Chang C C, Lin C J. Libsvm: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology (TIST), 2011, 2(3): 1-27

[60] Rodriguez P, Bautista M A, Gonzalez J, et al. Beyond one-hot encoding: Lower dimensional target embedding. Image and Vision Computing, 2018, 75: 21-31

[61] De cock M, Dowsley R, Horst C, et al. Efficient and private scoring of decision trees, support vector machines and logistic regression models based on pre-computation. IEEE Transactions on Dependable and Secure Computing, 2017, 16(2): 217-230

[62] Bogatov D, Kollios G, Reyzin L. A comparative evaluation of order-revealing encryption schemes and secure range-query protocols. Proceedings of the VLDB Endowment, 2019, 12(8): 933-947

HAN Zhao-Yang, Ph. D. candidate. His current research interests include federated learning, and private computing.

GE Chun-Peng, Ph. D., professor. His research interests include searchable encryption and blockchain.

LIU Zhe, Ph. D., professor. His research interests include cryptography engineering and post-quantum cryptography.

FANG Li-Ming, Ph. D., professor. His research interests include searchable encryption and AI security.

Background

The research problem addressed in this paper lies within the field of federated learning and secure computation, with a focus on efficient prediction tasks for tree-based models. Federated learning has emerged as a promising approach for collaborative model training while preserving data privacy among multiple parties. However, existing research in this field primarily revolves around reusing training paradigms, resulting in additional overhead and limitations in terms of efficiency.

Internationally, the development of federated modeling schemes has provided significant advancements in collaborative machine learning, allowing multiple parties to train various models while withholding their private data. However, the need for efficient prediction tasks on these models has become increasingly prominent. Despite the progress made in federated learning, the challenge of achieving efficient and privacy-preserving prediction services for tree-based models, such as gradient boosting trees and random forests, remains unresolved to a large extent.

This paper aims to bridge this gap by proposing the OREC secure comparison protocol, specifically designed to enable secret comparison of confidential values with the assistance of an untrusted third party. The protocol leverages order-preserving encryption techniques to reveal the order relationship between ciphertexts through public functions, enhancing privacy while ensuring efficient and secure comparisons. Additionally, the integration of a one-time pad mechanism and obfuscated password sequences further strengthens the privacy preservation capabilities of the protocol. The research project to which this subject belongs is focused on ex-

ploring practical solutions for secure and efficient prediction tasks in federated learning. By addressing the limitations of existing approaches and introducing the OREC protocol, the project aims to facilitate the wider adoption of federated learning in real-world scenarios, where privacy and efficiency are crucial concerns. The significance of this project lies in its potential to advance the field of federated learning by offering an effective solution for secure and efficient prediction services in tree-based models. By providing a comprehensive security analysis, the protocol demonstrates its robustness against malicious third parties, further highlighting its practical utility.

This paper addresses the issues of privacy protection in AI and federated learning research, which are relevant to the project that supports this work. By introducing ORE-based technology, this paper expands the means of protecting AI models and explores the potential of applying privacy protection strategies from other domains to AI models. Furthermore, this paper extends the scope of federated learning applications by investigating the problem of federated inference for models after collaborative modeling and providing an efficient and reliable solution.