计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.241200115

隐私保护的决策树算法设计与应用

李进成　李英娜　付国庆
昆明理工大学信息工程与自动化学院　昆明 650500
云南省计算机技术应用重点实验室　昆明 650500
(2357599340@qq.com)

摘　要　在信息时代,数据成为一种宝贵资源。数据共享在驱动人工智能领域发展的同时,也带来了隐私泄露的风险。全同态加密(Fully Homomorphic Encryption,FHE)技术为各种机器学习算法的实现提供了一条安全路径,它允许在密文数据上直接进行运算。然而,在密文数据上进行运算会产生很高的计算开销,因此需要以“FHE友好”的方式重新设计算法。对此,基于CKKS全同态加密算法,采用低次近似的阶跃函数和轻量级的交互协议取代复杂的非线性运算,提出了一种新的隐私保护决策树方案,实现了密文下决策树的训练与推理。最后,在4个UCI数据集上进行了对比实验,实验结果显示,提出的方案在平均AUC和平均F1-Score指标上分别达到0.92与0.77,优于PrivaTree方案与SecDT方案,同时展现出更强的稳定性。

关键词:全同态加密;隐私保护;决策树
中图分类号　TP311

Design and Application of Decision Tree Algorithms for Privacy-preserving

LI Jincheng,LI Yingna and FU Guoqing
School of Information Engineering and Automation,Kunming University of Science and Technology,Kunming 650500,China
Yunnan Provincial Key Laboratory of Computer Technology Application,Kunming 650500,China

Abstract　In the digital era,data has emerged as a critical asset. Data sharing not only fuels advancements in the artificial intelligence sector,but also poses the threat of privacy violations. Fully Homomorphic Encryption(FHE) technology offers a secure solution for executing various machine learning algorithms on encrypted data,bypassing the risks associated with data exposure. Nonetheless,operations on encrypted data demand a significant computational overhead,prompting the need for algorithms to be redesigned with FHE optimization in mind. This paper introduces a novel privacy-preserving decision tree scheme based on the CKKS fully homomorphic encryption algorithm. It utilizes a low-degree approximate step function and a lightweight interaction protocol to supplant complex nonlinear operations,enabling the training and inference of decision trees directly on encrypted data. Extensive experiments on four benchmark UCI datasets reveal that the proposed scheme achieves an average AUC of 0.92 and an average F1-Score of 0.77,outperforming both the PrivaTree and SecDT schemes while also exhibiting greater stability.

Keywords　Fully homomorphic encryption,Privacy-preserving,Decision tree

1　引言

在人工智能的浪潮中,产品和服务搜集的数据已成为推动技术发展的关键动力。尤其是在智能互联网时代,端点设备收集并分析用户的行为数据以及用户与产品的互动信息,以实现产品智能化和提供定制化的用户体验。为了实现这些功能,存储在终端设备中的个人数据需要被多个服务器访问和处理,这种处理方式虽然能够极大地提升用户体验和满意度,但也不可避免地增加了数据隐私泄露的风险[1-2]。

近年来,严重的公共数据泄露事件频发,这些事件不仅侵犯了个人的隐私和安全,还威胁到了社会的稳定性。因此,必须采取严格的措施来保障个人隐私数据的安全,并且需要加强对第三方服务提供商管理的数据的保护。

随着技术的不断进步,目前已经开发出多种方法来保护隐私数据的安全,以尽量减少数据泄露的潜在风险。最基本的策略是对静态数据进行加密处理,以此确保即便数据库遭到非法窃取,其中的数据依旧是安全的。但是,随着云服务的快速发展,越来越多的应用程序能够直接访问到明文数据(即未经加密的数据),这导致传统的加密手段在提供安全性方面可能变得不够可靠[3]。理想情况下,我们期望所有应用程序都能在无需访问明文数据的情况下正常运行。同态加密技术,尤其是全同态加密技术,为实现这一理想状态提供了技术可能性,它允许在加密数据上直接进行计算,而无需将数据解密,这大大增强了在数据使用性和隐私保护之间的平衡。

全同态加密技术[4-7]使得可以在不解密的情况下对加密数据(即密文)进行运算,这种运算称为“同态计算”。它保证

基金项目:云南省重大专项计划(202302AD080002,202402AD080003)
This work was supported by the Major Science and Technology Project of Yunnan(202302AD080002,202402AD080003).
通信作者:李英娜(liyingna@kust.edu.cn)

了计算结果与在未加密数据(即明文)上进行相同运算的结果是一致的。具体来说,全同态加密支持对密文数据执行加法和乘法操作,从而使得任意复杂度的多项式计算成为可能。然而,FHE 的一个主要弊端是进行乘法运算时需要付出巨大的代价,这包括计算时间和所需的存储空间。因此,实际运用中对参与同态加密计算的多项式的阶数和乘法的次数是有一定限制的,目的是确保操作的可行性和效率[8]。

以往关于 FHE 和机器学习的交叉研究主要集中在逻辑回归模型[9-10]以及对浅层神经网络的研究[11]。虽然这两类模型在各领域中有广泛的应用,但它们远未涵盖所有常用的机器学习模型。在实际应用中,基于树的模型是最受欢迎的方法之一,如单个决策树、随机森林以及提升方法。

决策树是一种用于预测的模型,即将特征向量映射到分数或标签。其通过从根遍历到叶子节点以完成预测,路径由一系列"$x_i>\theta$"的比较操作确定,其中x_i是特征值,θ是阈值(如果满足条件则继续蔓延至右子树节点,否则蔓延至左子树节点)。

任何针对加密数据的决策树模型都需解决一个最大的问题:如何对加密数据进行对比操作。通常的做法是设计一种交互协议[12],并且对其进行实例化,但会产生与树的大小成比例的通信复杂性。在近期的工作中[13-15],其通信复杂性与树的大小或深度成比例,这对通信带宽造成了重大负担,从而出现了急需解决的问题:如何在加密数据上训练决策树,以及如何在加密数据上使用决策树进行轻量级通信预测。

鉴于上述背景,本文构建的假设场景聚焦于企业环境,旨在设计一种既符合现有企业架构又能提供隐私保障的决策树方案。在这一设定中,企业拥有多个数据源,并且会持续地将新数据上传到"数据湖"以便存储。这些数据在数据湖中采用加密形式保存,并且被众多微服务端(即微服务器)所调用。这些微服务端负责为企业提供密钥管理服务(Key Management Services,KMS)。考虑到企业数据的敏感性,KMS 的安全性必须得到严格的保障。同时,出于技术实施考虑,KMS 应当设计为轻量级的。本研究旨在确保微服务端无法接触到明文数据,并能在加密数据上执行计算,同时生成加密结果,这些结果随后可以通过 KMS 进行解密。这样的研究目标在保护了数据隐私的同时,也满足了企业对技术可行性的需求。

2 相关工作

隐私保护决策树算法根据算法阶段可分为预测阶段的隐私保护决策树算法和训练阶段的隐私保护决策树算法。

在"预测阶段"的相关工作中,部分研究提出了与决策树的规模[13-14,16]或深度[17]相关的交互式通信协议。与此相对照的是,Lu 等[18]和 Tueno 等[15]各自提出了非交互式的协议。这些协议的通信复杂度都随着树的规模或深度线性增加,并且仅支持低精度数据处理,要求对数据进行特定编码以及在有限域上进行同态计算。然而,这些研究所采用的特殊数据编码方式与本文所设想的应用场景并不相符。在现实企业环境中,数据通常以文本形式展现,存储于数据湖中,并可供多种微服务端访问。

此外,在 Tueno 等的工作中,还提出了一个预测协议,该协议将输入数据以二进制形式处理,并在决策树的每个节点上执行逐位比较。该协议实例化的乘法深度随输入数据位数线性增长。Tueno 等在一个 16 线程的计算机上对深度为 3 的树进行了性能测试,单个样本的平均计算时间在使用 TFHE 和 HElib 库时分别为 0.188 s 和 8.122 s。他们还对使用 HElib 库的算法进行了批处理评估,在同样 16 线程计算机上,平均计算时间为 10 ms。这一结果突显了优化算法和使用并行计算资源在提高效率方面的潜力。

在"训练阶段"的相关工作中,早期的研究[19-22]主要关注多方计算问题,即多个参与方在保护各自私有数据集不被泄露的前提下,通过实时通信共同训练模型。然而,这些方法的通信复杂性往往与数据集的大小成正比,存在通信效率的问题。与之不同,本文聚焦于"企业"这一特定情景,在此情境下,所有数据均为加密形态,除了企业自身,没有任何一方能够接触到明文数据。更重要的是,本文提出的协议设计克服了通信复杂性与数据集规模直接相关的问题,其通信复杂度并不随数据集的大小变化而变化,这一特点显著提升了协议在实际企业环境中的适用性和效率。

在模型的"训练阶段"与"预测阶段",都不可避免地会涉及到复杂计算的优化问题,如激活函数的近似计算等,旨在降低整体计算复杂度。以往的研究尝试通过多种方式近似常见的激活函数,例如,采用多项式函数对 Sigmod,ReLU 以及 Tanh 等激活函数进行逼近[9,23-24],或者应用费马小定理近似阶跃函数[15-18]。然而,这些方法并不符合本文所面临的研究场景的需求。

针对本文的研究背景,更为相关的方法是 Cheon 等提出的基于实数域的低阶函数逼近方法[13]。该方法通过在一定阈值范围内使用 l_∞ 范数进行函数逼近,得以实现最佳的渐进复杂度。本研究则采用 l_2 范数进行逼近,相比之下,计算时间减少了 45%。对于这一成果的具体分析和探讨,将在第 7 章详细阐述。

3 预备知识

本章将详细说明本文中使用的专业术语和符号,以及统一收敛、决策树、CPA(Chosen Plaintext Attack,CPA)-安全、CKKS 全同态加密方案和隐私保护协议的标准定义。

3.1 专业术语与符号定义

本文涉及的主要符号如表 1 所列。

表 1 符号定义

Table 1 Definition of symbols

符号	说明
$[n]$	对于 $n\in\mathbb{N}$,使用$[n]$表示集合$\{1,2,\cdots,n\}$
$\boldsymbol{x}=(x_1,x_2,\cdots,x_L)$	一个 L 维的二进制向量$\boldsymbol{x}=(x_1,x_2,\cdots,x_L)$,如果其第 ℓ 项是唯一的非零项,则其被称为 $\ell\in[L]$的 1-*hot* 编码
$\boldsymbol{v}_l$	向量 $\boldsymbol{v}$ 的第 ℓ 个分量
$v[l]$	向量 $\boldsymbol{v}$ 的第 ℓ 个分量
$neg(\cdot)$	一个不需要指定名称的函数
λ	安全参数,其取值根据所需的安全等级设定,通常取值为 128,256 或者 512

（续表）

符号	说明
PPT	即 Probabilistic Polynomial Time，表示概率多项式时间
Pr[]	某事发生的概率
ε	一个公钥加密方案，$\varepsilon=(Gen,Enc,Dec)$
v	一棵决策树中的一个节点
$v.feature$	一棵决策树中节点 v 的特征
$v.\theta$	一棵决策树中节点 v 的阈值
$v.left$	一棵决策树中节点 v 的左子节点
$v.right$	一棵决策树中节点 v 的右子节点
$v.leaf\_value$	一棵决策树中节点 v 的标签

定义一个概率集合 $X=\{X(a,n)\}_{a\in\{0,1\}^*,n\in\mathbb{N}}$ 是一个由 $a\in\{0,1\}^*$ 和 $n\in\mathbb{N}$ 决定的无限序列。定义两个无限序列 $X=\{X(a,n)\}_{a\in\{0,1\}^*,n\in\mathbb{N}}$ 和 $Y=\{Y(a,n)\}_{a\in\{0,1\}^*,n\in\mathbb{N}}$，称 X 和 Y 是计算不可区分的，记为 $X\approx_c Y$。如果满足所有非一致多项式时间算法 $\mathcal{D}$，存在可忽略的函数 neg，使得每个 $a\in\{0,1\}^*$ 和 $n\in\mathbb{N}$，都有：

$$|\Pr[\mathcal{D}(X(a,n))=1]-\Pr[\mathcal{D}(Y(a,n))=1]|\leqslant neg(n) \tag{1}$$

3.2 一致收敛

本文中使用函数收敛的概念：

设 $S(x)$ 为 $\sum_{n=1}^{\infty}u_n(x)$ 在区间 I 上的和函数，若对任意给定的 $\varepsilon>0$，都存在一个 N，使得当 $n>N$ 时，对区间 I 上的一切 x 都有：

$$|S(x)-S_n(x)|<\varepsilon \tag{2}$$

3.3 CPA-安全公钥加密体制

公钥加密方案具有以下语法和正确性要求。

公钥加密方案（Public Key Encryption，PKE）：一个完整的公钥加密方案具有以下 3 个算法：

（1）Gen（密钥生成）：以安全参数 1^λ 作为输入，输出（pk，sk），即公钥 pk 和密钥 sk，表示为 $Gen(1^\lambda)\to(pk,sk)$。

（2）Enc（加密）：以明文 $m\in\mathcal{M}$ 和公钥 pk 作为输入，输出密文 c，表示为 $Enc_{pk}(m)\to c$。

（3）Dec（解密）：以密文 c 和密钥 sk 作为输入，输出明文 $m'\in\mathcal{M}$，表示为 $Dec_{sk}(c)\to m'$。

正确性：对于某方案，如果在 $Gen(1^\lambda)$ 中，对于每对密钥对（pk，sk）和每个明文 $m\in\mathcal{M}$，都有：

$$\Pr[Dec_{sk}(Enc_{pk}(m))=m]=1 \tag{3}$$

则该方案被称为是正确的。

对于一个公钥加密方案 $\varepsilon=(Gen,Enc,Dec)$，如果没有敌手 $\mathcal{A}$ 可以在 *PPT* 内区分两个等长明文 x_0，x_1 的加密，则称 ε 是 *CPA* 安全的。正式的证明如下：

定义 *CPA* 抗撞击实验——$EXP^{cpa}_{\mathcal{A},\varepsilon}(\lambda)$：

（1）$\mathcal{B}$ 使用 $Gen(1^\lambda)$ 生成密钥对（pk，sk）。

（2）$\mathcal{B}$ 向 $\mathcal{A}$ 提供公钥 pk 与加密算法 $Enc_{pk}(\cdot)$，A 向 B 发送两个等长明文 $x_0,x_1\in\mathcal{M}$。

（3）$\mathcal{B}$ 选择一个随机比特 $b\in\{0,1\}$，而后计算密文 $c\leftarrow Enc_{pk}(x_b)$ 并发送给 $\mathcal{A}$。

（4）A 输出一个比特 b'。

（5）如果 $b=b'$，则该实验的输出定义为 1（否则定义为 0）。

CPA 安全：对于一个公钥加密方案 $\varepsilon=(Gen,Enc,Dec)$，如果任意敌手 $\mathcal{A}$ 在 *PPT* 内存在如下可忽略的函数 neg，使得：

$$\Pr[EXP^{cpa}_{\mathcal{A},\varepsilon}(\lambda)]\leqslant\frac{1}{2}+neg(\lambda) \tag{4}$$

则称 ε 是 *CPA* 安全的。

3.4 CKKS 全同态加密方案

CKKS 全同态加密方案是 Cheon 等基于 RLWE（Ring-Learning With Error，RLWE）问题，提出的一种支持近似计算的全同态加密方案。该方案的核心在于将同态加密过程中引入的噪声看作近似计算过程中误差的一部分，即有：对于明文消息 m，使用密钥 sk 加密得到密文 c，拥有解密结构 $\langle c,sk\rangle=m+e\approx m(\mathrm{mod}\ q)$。对于一个 2 的整数幂 N，将 N 维分圆多项式环记为 $R=\mathbb{Z}[X]/(X^N+1)$。对于一个正整数 q，将 R 模 q 上的剩余环记为 $R_q=\mathbb{Z}[X]_q/(X^N+1)$。CKKS 全同态加密方案的主要内容如下：

（1）生成密钥 $Gen(1^\lambda)$：根据安全参数 λ，生成公钥 pk 和密钥 sk，计算密钥 evk。

（2）加密 $Enc(m,pk)$：输入明文多项式 $m\in R$，输出对应的密文多项式 c_m。

（3）解密 $Dec(c_m,sk)$：输入密文多项式 $c_m\in R_q^2$，输出对应的明文多项式 m。

（4）加法 $Add(c_a,c_b)$：输入 $a,b\in R$ 的密文多项式 $c_a,c_b\in R_q^2$，输出 $a+b$ 的密文多项式 c_{a+b}。

（5）减法 $Sub(c_a,c_b)$：输入 $a,b\in R$ 的密文多项式 $c_a,c_b\in R_q^2$，输出 $a-b$ 的密文多项式 c_{a-b}。

（6）密文-密文乘法 $Mult(c_a,c_b,evk)$：输入 $a,b\in R$ 的密文多项式 $c_a,c_b\in R_q^2$，输出 $a\cdot b$ 的密文多项式 $c_{a\cdot b}$。

（7）明文-密文乘法 $CMult(m,c_a,evk)$：输入 $m\in R$ 的明文多项式和 $a\in R$ 的密文多项式 $c_a\in R_q^2$，输出 $m\cdot a$ 的密文多项式 $c_{m\cdot a}$。

（8）重缩放 $Rescale(c_a,p)$：输入 $a\in R$ 的密文多项式 $c_a\in R_q^2$ 和 2 的整数幂 p，输出 $p^{-1}\cdot m$ 的密文 $c_{p^{-1}\cdot m}$。

（9）旋转 $Rotate(c_a,r)$：输入 $a\in R$ 的密文多项式 $c_a\in R_q^2$ 和旋转次数 $r\in Z$，输出 c_a 旋转 r 次后的密文多项式 $c_a'\in R_q^2$。

3.5 隐私保护的两方协议

接下来，本文将详细介绍安全性的定义，并阐明在哪些特定条件下，客户端-服务器协议能够对抗半诚实服务器的攻击，以确保隐私保护。所谓“半诚实服务器”，是指那些遵守协议执行过程的服务器，但它可能会试图通过分析加密传输中的密文，来获取客户端额外的隐私信息。在此基础上，我们将评估协议在半诚实模型下的安全性，确保即使服务器尝试进行窥探，客户端的隐私数据也能得到有效保护。

本文协议涉及两方，即客户端和服务器，分别用 *Client* 和 *Server* 表示，其中客户端具有输入 x，服务器没有输入（表示为输入 $\perp$），并且两者都有安全参数 λ。客户端和服务器在

交互协议($\pi=\langle Client, Server\rangle$)中进行交互操作。在客户端输入 x、服务器的输入 $\perp$ 和安全参数 λ 上执行该协议,记为 $\langle Client(x), Server\rangle$。

对于一个交互协议 $\langle Client, Server\rangle$,如果 *Client* 和 *Server* 是在 *PPT* 内并且满足以下条件:

(1)完备性:存在一个可忽略的函数 *neg* 使得对于所有 $\lambda\in\mathbb{N}, x\in A$,有:

$$\Pr[out^{\pi}_{Client}(x,\perp,\lambda)=F(x)]=1-neg(\lambda) \tag{5}$$

(2)隐私性:对于每一个在 PPT 内的区分器 D,选择了等长明文 $x_0, x_1\in A$,存在一个可忽略的函数 *neg* 使得对于所有 $\lambda\in\mathbb{N}$,使得:

$$|\Pr[\mathcal{D}(view^{\pi}_{Server}(x_0,\perp,\lambda))=1]-\Pr[\mathcal{D}(view^{\pi}_{Server}(x_1,\perp,\lambda))=1]|\leqslant neg(\lambda) \tag{6}$$

则称这个交互协议是隐私保护的。

4 带有近似阶跃函数的决策树算法

本章将介绍用于训练和预测的决策树算法,该算法的核心在于:使用一个低次的多项式逼近决策树中的阶跃函数"$x<\theta$",以降低同态计算的复杂性。

为了构造一个低次的多项式逼近阶跃函数,先从最简单的一个情景考虑:阈值为 0 的阶跃函数。不妨设:

$$I_0(x):=\begin{cases}1, & x\geqslant 0\\ 0, & x<0\end{cases} \tag{7}$$

这里使用均方积分方法,即将 *soft-setp* 函数转换为以下优化问题的解:

$$\Phi=\min_{p\in P_n}\int_{-5}^{5}(I_0(x)-p(x))^2\mathrm{d}x \tag{8}$$

其中,P_n 是次数不超过实数 n 的多项式函数集合。

我们在对数据进行标准化后,将数据范围约束在区间 $[-1,1]$ 上,因此,将近似区间设置为 $[-5,5]$ 是足够的,过小的区间会导致误差过大,而过大的区间会增加计算的复杂度。

然而,在大多数的情况下,阶跃函数的近似误差在整个区间上是不均匀分布的。特别是在阈值附近,近似误差往往是最小的。因此,为减少这种情况导致的近似误差,本文引入权重函数,将优化问题转化为以下问题:

$$\Phi=\min_{p\in P_n}\int_{-5}^{5}(I_0(x)-p(x))^2 w(x)\mathrm{d}x \tag{9}$$

其中,对于 $x\in[-5,5]$,$w(x)>0$,并且 $\int_{-5}^{5}w(x)\mathrm{d}x=1$。

对于上述优化问题,常见的方法是使用 l_∞ 范数进行逼近,即 Remez 算法。然而,本文使用均方误差来对上述问题进行求解,因为 Remez 算法会出现数值不稳定的现象,进而导致多项式出现"大幅度的震荡"。

5 隐私决策树算法的训练阶段与预测阶段

本章提出了基于树模型的预测算法与隐私保护协议,以及训练算法与隐私保护协议。在这些算法和隐私保护协议中,所有参与计算的数据均由 FHE 算法进行加密。

5.1 预测阶段

我们提出了基于预测阶段的决策树算法与隐私保护协议。基于预测阶段的决策树算法见算法 1 和算法 2,在算法 1 中,我们用 *soft-step* 函数 $\Phi(x[v.feature]-v.\theta)$ 替换 *step* 函数 $I_{v.\theta}(x[v.feature])$;而后,遍历决策树中的所有路径,并计算所有叶子节点值的加权组合,其中每个叶子节点的值是与叶子节点标签的 1-*hot* 编码;最后计算得到一个 L 维向量,其中各个分量对应一个标签的概率值,并输出概率值最高的标签。算法 2 则是针对不是叶子结点的情况。

算法 1 预测阶段的决策树算法

假设 T 是一棵决策树,其中每个节点 v 包括 v.feature(特征)和 v.θ(阈值)。v.leaf_value 是与 v 相关的标签的 1-hot 编码值;v.left 和 v.right 分别代表 v 的左子节点和右子节点;v 被初始化为 T 的根节点。

输入:x∈[−1,1]^k,其中 k 是特征的数量

输出:y∈{0,1}

```
function DecisonTree_Predict(v,x)
If v is a leaf then
    returnv.leaf_value
else
    return Φ(x[v.feature]−v.θ)·DecisonTree_Predict(v.right,x)+
    Φ(v.θ−x[v.feature])·DecisonTree_Predict(v.left,x)
    end if
end function
```

对于预测阶段决策树的隐私保护协议,该协议只考虑两方:持有数据的客户端和持有(决策树)模型的服务器端。该协议中,通信复杂性与输入数据的大小呈比例关系,而不受决策树的大小和深度影响。该协议能够适用于各种基于树的预测模型,如随机森林算法,并在数据与模型均被加密的前提下,确保双方的安全性。具体如协议 1 所示。

算法 2 预测阶段的决策树子算法

输入:决策树 T 的一个节点 v,L 维的密文数据 c

输出:结果 result

```
function Sub_DecisionTress_Predict(v,c)
    if v 不是叶子结点 then
        return
            result=Φ(c[v.feature]−v.θ)·Sub_DecisonTree_Predict
            (v.right,x)+Φ(v.θ−c[v.feature])·Sub_DecisonTree_
            Predict(v.left,x)
        else
            return result=v.leaf_value
    end if
end function
```

协议 1 预测阶段决策树的隐私保护协议

客户端输入:数据 x∈[−1,1]^k

客户端输出:标签 l∈L

协议 PP=⟨C_{PP},S_{PP}⟩的具体流程如下:

1. 输入阶段:

   1.1. C_{PP} 使用 Gen(1^λ)生成公钥密钥对(pk,sk)。

   1.2. C_{PP} 将明文数据 x 加密得到密文数据 c,并将 c 和 pk 发送给 S_{PP}。

2. 计算阶段:

   S_{PP} 接收到 c 后,从根节点开始按照算法 2 进行递归计算,在根节点完成计算后,返回 L 维的密文数据 result。

3. 输出阶段:

   S_{PP} 将 result 发送给 C_{PP},C_{PP} 使用 sk 进行解密得到 m_{result},并得到标签 $label\leftarrow\arg\max_{l\in L}(m_{result})_l$

在协议 1 中,一棵决策树 T 被加密为:对于 T 中的每个节点 v,首先将 $v.feature$ 转换为 1-*hot* 编码;然后使用 pk 将 $v.feature$,$v.\theta$ 以及 $v.leaf\_value$ 进行加密,得到 $\tilde{v}.feature$,

$\tilde{v}.\theta$ 以及 $\tilde{v}.leaf\_value$，而不需要对 $v.left$ 和 $v.right$ 加密，因为它们并不涉及到模型的核心，特别需要注意，在协议 1 的第 2 步中，我们使用加密的 $v.feature$ 的 1-*hot* 编码，而不是直接使用 $v.feature$ 进行计算。最后，返回 $v.leaf\_value$ 的密文数据，而不是明文数据。

此外，在一些其他的应用中，若需要将用于预测的树模型进行加密保护，仅需要对协议 1 进行参数上的微调，即可将其转换为一个满足加密数据与加密模型的计算协议。

5.2 训练阶段

本节将介绍基于训练阶段决策树的算法和隐私保护协议。在决策树的训练算法中，使用式(9)可以得到一个低次多项式用于逼近阶跃函数，具体如算法 3 所示。隐私保护协议如协议 2 所示。

算法 3 训练阶段的决策树算法

这里使用$(\mathcal{X},\mathcal{Y})$代表参与计算的数据集，使用 $W=\{w_x\}_{x\in\mathcal{X}}$代表一组初始化为 1 的权重；使用 k 和 L 分别代表特征和标签的数量；使用 S 代表所有阈值的集合；使用 Φ 代表根据式(9)得到的低次多项式；参数 max_depth 代表决策树的深度，depth 初始化为 0。

输入：n 个样本$\mathcal{X}$及其相对应的标签$\mathcal{Y}$，其中每个样本 $x\in\mathcal{X}$的值域为 $[-1,1]^k$，标签$y_x\in\mathcal{Y}$是 1-hot 编码

输出：一颗深度为 max_depth 的决策树 T=(V,E)，其中每个节点 $v\in V$ 包含 v.feature(特征)，v.θ(阈值)，v.leaf_value(标签)。如果 v 是叶子结点，则 v.left 和 v.right 分别代表其左子节点和右子节点

1. function DecisionTree_Train($(\mathcal{X},\mathcal{Y})$,W,depth,v)
2. if 达到了最大深度 max_depth then
3. $v.leaf\_value\leftarrow \arg\max_{\ell\in[L]}\sum_{x\in\mathcal{X}} w_x\cdot y_x$
4. else
5. for 每个特征 i 和阈值 θ do
6. $right[i,\theta]\leftarrow \sum_{x\in\mathcal{X}} Mult(CMult(w_x\cdot y_x)\cdot\Phi(Sub(x[i]-\theta)))$;
7. $left[i,\theta]\leftarrow \sum_{x\in\mathcal{X}} Mult(CMult(w_x\cdot y_x)\cdot\Phi(Sub(\theta-x[i])))$;
8. end for
9. $(v.feature,v.\theta)\leftarrow Gini(\{right[i,\theta],left[i,\theta]\}_{i\in[k],\theta\in S})$;
10. $w_x^{right}\leftarrow CMult(w_x\cdot\Phi(Sub(x[i^*]-\theta^*)))$;
11. DecisionTree_Train((X,Y),$\{w_x^{right}\}_{x\in\mathcal{X}}$,depth+1,v.right);
12. $w_x^{left}\leftarrow CMult(w_x\cdot\Phi(Sub(\theta^*-x[i^*])))$;
13. DecisionTree_Train((X,Y),$\{w_x^{left}\}_{x\in\mathcal{X}}$,depth+1,v.left);
14. end if
15. end function

协议 2 训练阶段决策树的隐私保护协议

客户端输入：n 个样本$\mathcal{X}$及其相对应的标签$\mathcal{Y}$，其中每个样本 $x\in\mathcal{X}$的值域为$[-1,1]^k$，标签$y_x\in\mathcal{X}$是 1-hot 编码。

客户端输出：一棵深度为 max_depth 的决策树 T=(V,E)，其中每个节点 $v\in V$ 包含 v.feature(特征信息)，v.θ(阈值)，v.leaf_value(标签信息)。如果 v 是叶子结点，则 v.left 和 v.right 分别代表其左子节点和右子节点。

协议 TP=$\langle C_{TP},S_{TP}\rangle$的具体流程如下：($[\![\cdot]\!]$代表加密后的密文数据)

1. 输入阶段：
   1.1. C_{TP}使用 $Gen(1^\lambda)$生成公钥密钥对(pk,sk)。
   1.2. C_{TP}使用 pk 加密每个样本 $x\in X$ 及其标签$y_x\in\mathcal{Y}$，得到相对应的密文数据c_x和c_{y_x}，并将$\{c_x,c_{y_x},pk\}$发送给S_{TP}。
2. 计算阶段：
   对于每个节点 v，这里使用$W_v=\{[\![w_x]\!]\}_{x\in\mathcal{X}}$代表与 v 对应的权重，将其初始化为$[\![1]\!]$，并将 depth 初始化为 0；对于 depth=1,2,…，max_depth 的节点 v，通过$S_{TP}=(c_x,c_{y_x},pk,W_v,depth,v)$和$C_{TP}=$pk 联合计算得到$[\![v.feature]\!]$，v.θ，$[\![W_{v.right}]\!]$以及$[\![W_{v.left}]\!]$。(当 v 是叶子节点时，还将计算得到 v.leaf_value)
3. 输出阶段：
   S_{TP}将训练后的加密决策树发送给C_{TP}，C_{TP}使用 sk 将其解密，得到明文的决策树 T=(V,E)。

在协议 2 的计算阶段，服务器端(S_{TP})基于算法 3 进行同态运算，同时客户端(C_{TP})协助处理部分任务，从而降低了整体计算复杂度，实现了轻量级的计算。具体地，我们实现了密文数据上的同态运算(其中 *Gini* 指数的计算是在客户端的协同下完成的)。为了计算 *Gini* 指数，服务器首先同态地将训练集中的密文数据聚合为 $|S|\cdot k\cdot L$(其中 $|S|$ 是阈值的数量，k 是特征的数量，L 是标签的数量)，并将聚合结果发送给客户端。随后，客户端将这些阈值和特征重新加密，并把密文发送回服务器端，以便进行下一轮的协同计算。这种方法确保了客户端的计算负担不会因数据集的增大而显著提升，同时服务器端也无法接触到任何敏感的明文信息，从而维护了数据的安全性。

6 安全性分析

定理 1(预测阶段的隐私保护安全性) $PP=\langle Client_{PP},Server_{PP}\rangle$是一个用于算法 1 中的隐私保护协议，前提是其所使用的加密方案 ε 是满足 *CPA*-安全的。

证明：令 $\varepsilon=(Gen,Enc,Dec,Eval)$表示协议 $PP=\langle Client_{PP},Server_{PP}\rangle$中所使用的全同态加密方案，并且假定 ε 是 *CPA*-安全的。

首先分析 *PP* 的复杂性，并证明其是 *PPT* 的。$Client_{PP}$在给定输入 $x\in[-1,1]^k$的情况下执行以下操作：执行一次 *Gen*，执行 k 次 *Enc*(对应 x 中的每个特征)，执行 L 次 *Dec*(对应输出结果p_{res}中的每个标签的权重)，以及计算 L 个值中的最大值。由于 *Gen*，*Enc*，*Dec* 的复杂度都是 $poly(\lambda)$，因此可以得出$Client_{PP}$是 *PPT* 的，并且其复杂度与加密输入和解密输出成正比。$Server_{PP}$对每个内部节点执行固定次数的基本同态操作(即加法和乘法)，以及对每个叶节点执行 $O(L)$次操作，每个基本同态操作的复杂度是 $poly(\lambda)$。因此可以得出结论：$Server_{PP}$是 *PPT* 的，其复杂度为$O(m\cdot L)\cdot poly(\lambda)$。

接下来证明 $PP=\langle Client_{PP},Server_{PP}\rangle$是完备的。可以注意到 *PP* 同态地评估了与算法 1 中计算的同一个函数，因此从加密方案 ε 的正确性立即得出完备性。

最后，证明 $PP=\langle Client_{PP},Server_{PP}\rangle$满足 3.5 节隐私保护协议中的隐私条件。假设 *PP* 不满足隐私保护，即存在一个 *PPT* 内的区分器 D，其选择了两个等长明文$x_0,x_1\in A$，以及一个多项式$p(\cdot)$，对于无穷多个 $\lambda\in\mathbb{N}$，使得：

$$Pr[\mathcal{D}(view^{pp}_{Serverpp}(x_1,\perp,\lambda))=1]-P[\mathcal{D}(view^{pp}_{Serverpp}(x_0,\perp,\lambda))=1]\geqslant\frac{1}{p(\lambda)}\tag{10}$$

以下证明对于给定的 D，可以构造一个攻击者$\mathcal{A}$来破坏加密方案 ε 的 *CPA*-安全性。攻击者$\mathcal{A}$参与选择明文攻击$EXP^{cpa}_{\mathcal{A},\varepsilon}$的流程如下：

1)在接收 *pk* 后，执行$\mathcal{D}$以获得x_0,x_1，并将它们发送给$\mathcal{B}$。

2)在接收到$c_x \leftarrow Enc_{pk}(x_b)$后,行为要与$Server_{PP}$在执行$PP$时收到的来自$Client_{PP}$的$(c_x, pk)$后的行为完全一致。

3)在区分器$\mathcal{D}$上运行$view^{pp}_{Server_D}$并输出$\mathcal{D}$的结果。

由于可以对x_0, x_1进行高效的选取,以及$Server_{PP}$和$\mathcal{D}$都是PPT的,因此攻击者$\mathcal{A}$是PPT的。用$view^{cpa}_{Server_D}(x_{b^*}, \perp, \lambda)$来表示在执行$EXP^{cpa}_{\mathcal{A},\varepsilon}$的过程中,由挑战者选定比特$b^*$时,模拟由$\mathcal{A}$执行的$Server_{PP}$的视图。由于$\mathcal{A}$的行为与在$PP$中与$Server_{PP}$的完全一致,因此对于每个$b^* \in \{0,1\}$,都有:

$$\Pr[\mathcal{D}(view^{pp}_{Server_{PP}}(x_{b^*}, \perp, \lambda))=1] = \Pr[\mathcal{D}(view^{EXP^{cpa}}_{Server_{PP}}(x_{b^*}, \perp, \lambda))=1] \tag{11}$$

根据式(10)和式(11)可得:

$$\Pr[\mathcal{D}(view^{EXP^{cpa}}_{Server_{PP}}(x_1, \perp, \lambda))=1] - \Pr[\mathcal{D}(view^{EXP^{cpa}}_{Server_{PP}}(x_0, \perp, \lambda))=1] \geqslant \frac{1}{p(\lambda)} \tag{12}$$

故可得:

$$\begin{aligned}&\Pr[EXP^{cpa}_{\mathcal{A},\varepsilon}(\lambda)=1]\\ &=\frac{1}{2}\cdot(\Pr[EXP^{cpa}_{\mathcal{A},\varepsilon}(\lambda)=1|b=1]+\Pr[EXP^{cpa}_{\mathcal{A},\varepsilon}(\lambda)=1|b=0])\\ &=\frac{1}{2}\cdot(\Pr[\mathcal{D}(view^{EXP^{cpa}}_{Server_{PP}}(x_1, \perp, \lambda))=1]+\Pr[\mathcal{D}(view^{EXP^{cpa}}_{Server_{PP}}(x_0, \perp, \lambda))=1])\\ &\geqslant\frac{1}{2}+\frac{1}{2p(\lambda)}\end{aligned} \tag{13}$$

与假设矛盾,故满足隐私保护,证毕。

定理 2(训练阶段的隐私保护安全性) $TP=\langle Client_{TP}, Server_{TP}\rangle$是算法 3 的隐私保护协议,前提是其所使用的加密方案ε是满足CPA-安全的。

证明:令$\varepsilon=(Gen, Enc, Dec, Eval)$表示在协议$TP=\langle Client_{TP}, Server_{TP}\rangle$中使用的 FHE 加密方案,假设$\varepsilon$是 CPA 安全的。

首先分析TP的复杂性,可以证明它是PPT的。分别用$k, L, m, |S|$表示特征的数量、标签的数量、决策树节点的数量以及所考虑的阈值的数量,并且λ是安全参数。首先,分析$Server_{TP}$。在协议中,$Server_{TP}$对每个叶子以及每个内部节点和每个阈值和特征执行$n \cdot L$次同态乘法运算和n次同态加法运算,再加上额外k次同态乘法和加法运算,以处理来自$Client_{TP}$的回复,且每个同态运算都是关于安全参数λ的多项式时间的。因此,$Server_{TP}$是PPT的,总体复杂度为$O(n \cdot m \cdot |S| \cdot k \cdot L) \cdot poly(\lambda)$。而后分析$Client_{TP}$。$Client_{TP}$在计算阶段执行$O(|S| \cdot k \cdot L)$的$Dec(\cdot)$操作和$O(k)$的$Enc(\cdot)$操作,以及计算明文数据的$Gini$。计算每个$Dec(\cdot)$和$Enc(\cdot)$的时间均为关于安全参数$\lambda$的多项式级别的,计算明文数据的$Gini$的时间是$O(|S| \cdot k \cdot L)$的。因此,$Client_{TP}$在训练阶段的复杂度为$O(m \cdot |S| \cdot k \cdot L) \cdot poly(\lambda)$。此外,$Client_{TP}$的整体计算过程(包括生成密钥、加密输入和解密输出等)都是关于其输入和安全参数的多项式时间的,因此$Client_{TP}$是PPT的。最后分析了通信的复杂性。在每个节点传输了$O(|S| \cdot k \cdot L)$个密文,因此通信复杂度为$O(m \cdot |S| \cdot k \cdot L) \cdot poly(\lambda)$。

接下来证明$TP=\langle Client_{TP}, Server_{TP}\rangle$是完备的。可以注意到$\langle Client_{TP}, Server_{TP}\rangle$同态地计算了与算法 3 中相同的函数。因此,完备性直接来源于加密方案ε的正确性。

最后,证明TP满足 3.5 节隐私保护协议中的隐私条件。假设TP的隐私性不成立,即存在一个PPT的区分器$\mathcal{D}$,它选择了等长密文对$(x_0, y_0), (x_1, y_1) \in A$,以及一个多项式$p(\cdot)$,对于无穷多个$\lambda \in \mathbb{N}$,使得:

$$\Pr[\mathcal{D}(view^{TP}_{Servcer_{TP}}((x_1, y_1), \perp, \lambda))=1] - \Pr[\mathcal{D}(view^{TP}_{Servcer_{TP}}((x_0, y_0), \perp, \lambda))=1] \geqslant \frac{1}{p(\lambda)} \tag{14}$$

下文中展示对于给定的$\mathcal{D}$,可以构建一个攻击者$\mathcal{A}$,该攻击者破坏了加密方案ε的CPA-安全性。

攻击者$\mathcal{A}$以如下方式参与$EXP^{cpa}_{\mathcal{A},\varepsilon}$:

1)在接收pk后,执行$\mathcal{D}$以获得$(x_0, y_0), (x_1, y_1)$,并将它们发送给$Chal$。

2)在接收到$CTXT \leftarrow Enc_{pk}(x_b, y_b)$后,除了每条从$Server_{TP}$发往$Client_{TP}$的信息$[\![right[i,\theta]]\!], [\![left[i,\theta]]\!]$(其中$i \in [k], \theta \in S$)之外,$\mathcal{A}$的其它行为要与$Server_{TP}$在执行$TP$并从$Client_{TP}$接收到$ctxt=\{c_x, c_{y_x}\}$和$pk$时的行为完全一致。对于这些信息,攻击者$\mathcal{A}$的应答如下:$\mathcal{A}$均匀随机采样$i \leftarrow [k]$和$\theta \leftarrow S$,计算$Enc_{pk}(i, \theta)$,并将此密文发送给$Server_{TP}$。

3)在$view_{Servcer_{TP}}$上运行$\mathcal{D}$,并输出$\mathcal{D}$的结果。

攻击者$\mathcal{A}$是PPT的,这是因为$(x_0, y_0), (x_1, y_1)$可以高效地采样,且$Server_{TP}$和$\mathcal{D}$都是PPT的。假设TP'是一个与TP完全相同的协议,除了对$Client_{TP}$,其他查询都是通过加密一个随机采样的$(i, \theta) \leftarrow [k] \times S$来回答的。我们用$view^{EXP^{cpa}}_{Servcer_{TP}}((x_b, y_b), \perp, \lambda)$来表示在执行$EXP^{cpa}_{\mathcal{A},\varepsilon}$时,由$\mathcal{A}$模拟的$Server_{TP}$的视图。根据$TP'$的定义,对于每个$b \in \{0, 1\}$,都有:

$$\Pr[\mathcal{D}(view^{TP'}_{Servcer_{TP}}((x_b, y_b), \perp, \lambda))=1] = \Pr[\mathcal{D}(view^{EXP^{cpa}}_{Servcer_{TP}}((x_b, y_b), \perp, \lambda))=1] \tag{15}$$

此外,ε的CPA-安全性保证了服务器在TP和TP'中的视图在计算上是无法区分的。式(15)进一步有:

$$\Pr[\mathcal{D}(view^{TP'}_{Servcer_{TP}}((x_1, y_1), \perp, \lambda))=1] - \Pr[\mathcal{D}(view^{TP'}_{Servcer_{TP}}((x_0, y_0), \perp, \lambda))=1] \geqslant \frac{1}{p(\lambda)} - neg(\lambda) \tag{16}$$

再结合式(15)和式(16)可得:

$$\Pr[\mathcal{D}(view^{EXP^{cpa}}_{Servcer_{TP}}((x_1, y_1), \perp, \lambda))=1] - Pr[\mathcal{D}(view^{EXP^{cpa}}_{Servcer_{TP}}((x_0, y_0), \perp, \lambda))=1] \geqslant \frac{1}{p(\lambda)} - neg(\lambda) \tag{17}$$

故可得:

$$\begin{aligned}&\Pr[EXP^{cpa}_{\mathcal{A},\varepsilon}(\lambda)=1]\\ &=\frac{1}{2}\cdot(\Pr[EXP^{cpa}_{\mathcal{A},\varepsilon}(\lambda)=1|b=1]+\Pr[EXP^{cpa}_{\mathcal{A},\varepsilon}(\lambda)=1|b=0])\\ &=\frac{1}{2}\cdot(\Pr[\mathcal{D}(view^{EXP^{cpa}}_{Servcer_{TP}}((x_1, y_1), \perp, \lambda))=1]+\Pr[\mathcal{D}(view^{EXP^{cpa}}_{Servcer_{TP}}((x_0, y_0), \perp, \lambda))=1])\\ &\geqslant\frac{1}{2}+\frac{1}{2p(\lambda)}-neg(\lambda)\end{aligned} \tag{18}$$

与假设矛盾,故TP满足隐私保护,证毕。

7 实验结果

本文算法与协议基于开源库 SEAL 中的 CKKS[5]算法均使用 C++语言实现。实验环境:CPU 为 Intel Core i7-

13700KF，主频 3.40 GHz，16 核 24 线程，内存 64 GB 的计算机，操作系统为 Ubuntu 20.04。

本研究选取了 4 个经典的 UCI 数据集，分别为 Cancer，Digits，Iris，和 Wine，以评估提出的隐私保护决策树算法和协议的性能，数据集的基本介绍如表 2 所列。为了构建符合式(9)要求的低阶多项式，这些数据集需要进行标准化处理。实验评估将重点放在两个核心性能指标上：准确性和计算效率。在准确性方面，采用 F1-score 和 AUC 作为评价标准。在计算效率方面，将具体量化算法和协议的运算时间。这样的评估能够全面地反映出所提方法在实际应用中的性能表现。

表 2　数据集介绍

Table 2　Datasets introduction

数据集	样本数量	特征量	来源
Cancer	150	4	https://archive.ics.uci.edu/
Digits	1797	64	
Iris	569	30	
Wine	178	13	

7.1　评估指标

由于是分类问题，所以这里选择 F1-Score 与 AUC 作为评价指标。

$$F1=2\times\frac{Precision\times Recall}{Precision+Recall} \tag{19}$$

其中，$Precision$ 为精确率，$Recall$ 为召回率。

F1-Score 用于衡量二分类模型的精确度，它是模型精确率(Precision)和召回率(Recall)的调和平均数。

AUC 表示 ROC 曲线下的面积，值范围从 0 到 1。AUC 越接近于 1，表示模型的性能越好；AUC 越接近于 0.5，表示模型的性能越接近于随机猜测；AUC 小于 0.5 表示模型性能不如随机猜测。

7.2　基础方案

在本研究中，为了全面评估本文提出的方法的有效性和效率，我们选择了两种具有代表性的现有隐私保护决策树方案进行详细比较。

(1)PrivaTree[25]：由 Zein 等提出，这是一种专为分布式、水平划分的生物医学数据集设计的高效且隐私保护的协议。PrivaTree 通过联邦学习，允许各数据持有者在不共享原始数据的前提下，计算并贡献对全局决策树模型的更新。这些更新在私有数据集上进行，并使用加法秘密共享技术进行隐私保护的聚合，以协作更新模型。实验结果表明，PrivaTree 在计算和通信效率上表现优异，生成的模型在准确性上只有适度损失，且明显优于各自独立训练的模型。

(2)SecDT[26]：由 Chen 等开发，这是一种基于保序加密机制的隐私保护决策树方案，应对垂直联邦学习环境中的隐私问题。它基于快速决策树(Very Fast Decision Tree，VFDT)技术，允许在数据连续到达的情况下逐步构建模型，同时通过保序加密保护训练过程中交换的数据统计信息，有效防止私人信息泄露。此外，通过区域计数的方法压缩统计数据大小，既保持了模型的准确性，也增强了隐私保护。广泛的实验评估显示，SecDT 在效率和隐私保护方面表现出色。

虽然两种方案均有效地解决了特定的隐私保护问题，但它们各有侧重，适用于不同的数据分布和协作环境。通过与这些方案的对比，本文方法展现了在某些关键方面的潜在优势。在后续的实验部分将详细展示这些优势，以及本文方法在实际应用中的表现。

7.3　对比结果

数据集采用 5 折交叉验证，为更全面客观地对比本文算法与两种方案的准确度，分别选取深度为 1，2，3，4，5 的决策树进行对比，并重复 10 次，结果如图 1 所示。

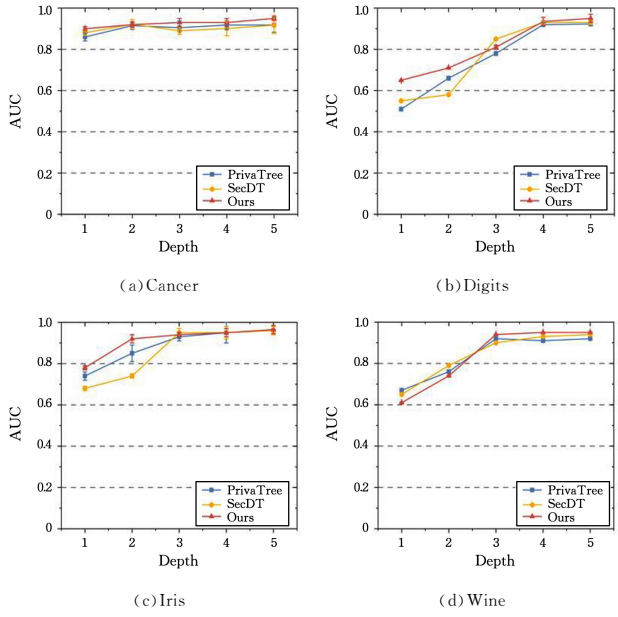


图 1　AUC 对比结果图

Fig. 1　AUC comparison result chart

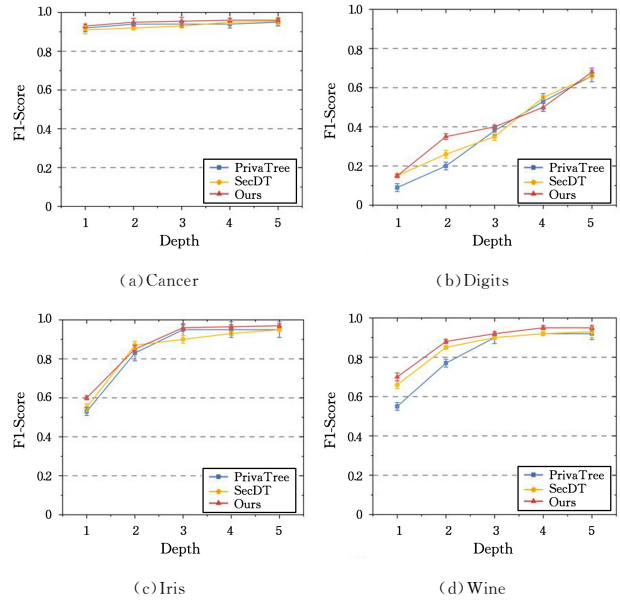


图 2　F1-Score 对比结果图

Fig. 2　F1-Score comparison result chart

由图 1 和图 2 可以看出，本文提出的算法(算法 1 和算法 3)在准确度(F1-Score 与 AUC)方面整体优于 PrivaTree 与 SecDT 方案，且具有更好的稳定性。具体地，本文算法在 4 个数据集上的平均 AUC 为 0.92，PrivaTree 方案为 0.845，SecDT 方案为 0.842；在平均 F1-Score 方面，本文的算法为 0.77，PrivaTree 方案为 0.741，SecDT 方案为 0.755。

总的来说，本文所提出的算法在准确度和稳定性方面，均优于 PrivaTree 与 SecDT 方案。

7.4 计算时间评估

这里选取 $depth=3$ 的决策树进行评估。在一棵决策树的计算过程中，训练过程往往占据较高比例的计算时间，这也是我们最关心的一项；其次便是预测过程，在该过程中，仅需对待预测的样本进行计算，因此耗时较短。实验结果如图 3、图 4 所示。

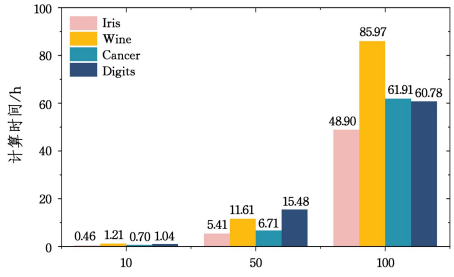


图 3 训练时间对比结果

Fig. 3 Training time comparison results

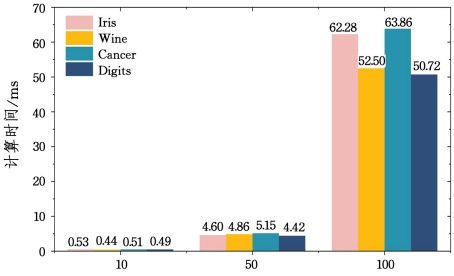


图 4 预测时间对比结果

Fig. 4 Forecast time comparison results

在密文数据集上进行训练时，这里将 $poly\_degree$ 设置为 8192，即每次可以将 4096 个数据嵌入多项式中；明文模的精度设置为 35 位。由于实验设备性能受限，在对 4 个数据集进行训练时，如果使用 10%的样本进行训练，则平均训练用时为 1 h；如果使用全部的训练样本，训练时间最长可达 85.97 h。在实际应用场景中部署时，可以考虑使用更强大的服务器来进行计算。另一方面，在预测阶段，即使是对整个测试集进行预测，所需的时间最多也仅为 63.86 ms，这样的计算速度无需商业级服务器即可轻松完成。

结束语 本文提出了一种隐私保护决策树方案，它包括一套算法和协议，能够对密文数据进行训练与预测。在训练阶段，我们采用轻量级协议进行必要的交互，显著降低计算时间；在预测阶段，则使用无需交互的协议，确保服务器端无法接触到明文数据，为客户端提供了高度的安全保障，同时保持了高计算效率。我们使用 4 个 UCI 数据集进行对比实验，实验结果显示，与 PrivaTree 和 SecDT 方案相比，本文算法在准确度(F1-Score 与 AUC)和稳定性方面表现得更加优秀。

参 考 文 献

[1] OLINDER N, FEDYAKIN K, KORNEEVA E. Personal data protection in the internet of things [C]// Proceedings of the 1st International Scientific Conference Legal Regulation of the Digital Economy and Digital Relations: Problems and Prospects of Development. New York, USA: Atlantis Press, 2021: 227-232.

[2] ZHANG J, CHEN B, ZHAO Y, et al. Data security and privacy-preserving in edge computing paradigm: Survey and open issues [J]. IEEE Access, 2018, 6: 18209-18237.

[3] SACHDEV A, BHANSALI M. Enhancing cloud computing security using AES algorithm [J]. International Journal of Computer Applications, 2013, 67(9): 19-23.

[4] BRAKERSKI Z. Fully homomorphic encryption without modulus switching from classical GapSVP [C]// Annual Cryptology Conference. Heidelberg, Germany: Springer, 2012: 868-886.

[5] CHEON J H, KIM A, KIM M, et al. Homomorphic encryption for arithmetic of approximate numbers [C]// Advances in Cryptology—ASIACRYPT 2017. Heidelberg, Germany: Springer International Publishing, 2017: 409-437.

[6] MASAHIRO Y. Fully Homomorphic encryption without bootstrapping [M]. Saarbrücken, Germany: LAP LAMBERT Academic Publishing, 2015.

[7] FAN J, VERCAUTEREN F. Somewhat practical fully homomorphic encryption [EB/OL]. [2012-03-22]. https://ia.cr/2012/144.

[8] ALLOGHANI M, ALANI M M, AL-JUMEILY D, et al. A systematic review on the status and progress of homomorphic encryption technologies [J]. Journal of Information Security and Applications, 2019, 48: 102362.

[9] CHEN H, GILAD-BACHRACH R, HAN K, et al. Logistic regression over encrypted data from fully homomorphic encryption [J]. BMC Medical Genomics, 2018, 11(4): 3-12.

[10] CROCKETT E. A low-depth homomorphic circuit for logistic regression model training [EB/OL]. [2024-05-01]. https://ia.cr/2020/1483.

[11] GILAD-BACHRACH R, DOWLIN N, LAINE K, et al. Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy [C]// International Conference on Machine Learning. New York, USA: PMLR, 2016: 201-210.

[12] BOST R, POPA R A, TU S, et al. Machine learning classification over encrypted data [EB/OL]. [2015-01-12]. https://ia.cr/2014/331.

[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 [J]. IEEE Transactions on Dependable and Secure Computing, 2019, 16(2): 217-230.

[14] ZHAO J, ZHU H, WANG F, et al. Efficient and privacy-preserving tree-based inference via additive homomorphic encryption [J]. Information Sciences, 2023, 650: 119480.

[15] TUENO A, BOEV Y, KERSCHBAUM F. Non-interactive private decision tree evaluation [C]// 34th Annual IFIP WG 11.3 Conference. Berlin, Germany: Springer International Publishing, 2020: 174-194.

[16] XU K, TAN B H M, WANG L P, et al. Privacy-preserving outsourcing decision tree evaluation from homomorphic encryption [J]. Journal of Information Security and Applications, 2023, 77: 103582.

[17] TUENO A, KERSCHBAUM F, KATZENBEISSER S. Private evaluation of decision trees using sublinear cost [J]. Proceedings on Privacy Enhancing Technologies, 2019(1): 266-286.

[18] LU W J, ZHOU J J, SAKUMA J. Non-interactive and Output Expressive Private Comparison from Homomorphic Encryption

[C]// Proceedings of the 2018 on Asia Conference on Computer and Communications Security. New York, USA: ACM, 2018: 67-74.

[19] HAN Z, GE C, WU B, et al. Privet: A privacy-preserving federated incremental decision trees [J]. IEEE Transactions on Services Computing, 2023, 16(3): 1964-1975.

[20] ZHENG Y, XU S, WANG S, et al. Privet: A privacy-preserving vertical federated learning service for gradient boosted decision tables [J]. IEEE Transactions on Services Computing, 2023, 16(5): 3604-3620.

[21] YAMAMOT F, OZAWA S, WANG L. eFL-Boost: Efficient federated learning for gradient boosting decision trees [J]. IEEE Access, 2022, 10: 43954-43963.

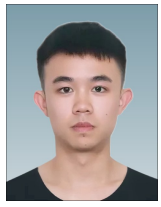
[22] ZHAO J, ZHU H, XU W, et al. SGBoost: An efficient and privacy-preserving vertical federated tree boosting framework [J]. IEEE Transactions on Information Forensics and Security, 2023, 18: 1022-1036.

[23] BLATT M, GUSEV A, POLYAKOV Y, et al. Optimized homomorphic encryption solution for secure genome-wide association studies [J]. BMC Medical Genomics, 2020, 13(7): 83.

[24] BOURA C, GAMA N, GEORGIEVA M, et al. Chimera: Combining ring-lwe-based fully homomorphic encryption schemes [J]. Journal of Mathematical Cryptology, 2020, 14(1): 316-338.

[25] EL ZEIN Y, LEMAY M, HUGUENIN K. PrivaTree: Collaborative privacy-preserving training of decision trees on biomedical data [J]. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2023, 21(1): 1-13.

[26] CHEN Y C, CHANG C C, HUNG C C, et al. SecDT: privacy-preserving outsourced decision tree classification without polynomial forms in edge-cloud computing [J]. IEEE Transactions on Signal and Information Processing over Networks, 2022, 8: 1037-1048.



LI Jincheng, born in 1998, postgraduate. His main research interests include cryptography and privacy protection.

LI Yingna, born in 1973, Ph.D supervisor. Her main research interests include big data analysis and industrial Internet.