Privacy-Preserving Decision Tree Solution in the 2-Part Fully Distributed Setting

Vu Thi Van, Luong The Dung, Hoang Van Quan, Tran Thi Luong, Hoang Duc Tho

***Abstract*— Data mining has emerged as an important technology for obtaining knowledge from big data. However, there are growing concerns that the use of this technology is infringing on privacy. This work proposes a decision tree mining solution according to the ID3 algorithm that ensures privacy in the 2-Part Fully Distributed setting.**

***Tóm tắt*— Khai phá dữ liệu đã nổi lên như một công nghệ quan trọng để thu thập kiến thức từ lượng dữ liệu khổng lồ. Tuy nhiên, ngày càng có nhiều lo ngại rằng việc sử dụng công nghệ này đang vi phạm quyền riêng tư của cá nhân. Bài báo này đề xuất giải pháp khai phá cây quyết định theo thuật toán ID3 có đảm bảo tính riêng tư trong mô hình phân tán đầy đủ hai bên.**

Keywords*— *Privacy-Preserving Data Mining; ID3; Decision tree; Elliptic curve.

Từ khóa*— *Khai phá dữ liệu có đảm bảo tính riêng tư; ID3; Cây quyết định; Đường cong Elliptic.

I. INTRODUCTION

Data mining is the process of extracting potentially valuable information from large amounts of data stored in databases or data warehouses. More specifically, it is the process of extracting, generating hidden, unknown, but useful knowledge or patterns from Big databases. Simultaneously, it is the process of generalizing discrete facts in data into generalized, regularisation knowledge that actively supports decision-making processes. However, due to legal constraints on privacy laws and information security policies of individuals and organizations, many organizations and individuals are not allowed to provide data sets for the mining process (for example personal data of customers in a bank, patient medical data...). As a result, the question is how to permit data mining on data sets while protecting the private information of individuals and organizations contained in the data. Solutions to this problem have been around since the 2000s, collectively known as Privacy-Preserving Data Mining (PPDM) [1].

Most PPDM techniques use some form of transformation on the original data to perform privacy protection [2]. There are two main approaches: randomization-based and cryptography-based.

These approaches are based on randomization techniques, such as additive data perturbation and random subspace projection, that mask the underlying data while preserving the statistical properties of the overall dataset. While these approaches are fast and efficient, they do not provide strong security guarantees and are often susceptible to attacks [3]. The solutions based on the perturbation approach are highly efficient but have a trade-off between privacy and accuracy, i.e., if we require more privacy, the miner loses more accuracy in the data mining results, and vice-versa [4].

The PPDM solutions based on cryptography typically consider the entire data (all attributes) as private and use cryptographic protocols such as homomorphic encryption, Yao's garbled circuits, etc. Most cryptographic-based approaches rely on peer-to-peer communication and are usually defined in 2-party scenarios, with extension to multi-party scenarios often resulting in significant communication overhead. For the PPDM solutions based on cryptography, the privacy of data holders is safely preserved and the output result is accurately guaranteed, but the performance is quite poor [5].

The decision tree algorithm is an algorithm commonly used in classification problems, such as letter classification in text recognition, etc. The ID3 decision tree algorithm (Iterative Dichotomiser 3) was born very early and is a widely used decision tree construction algorithm. In this work, we focus on privacy-preserving decision tree solution in the 2-Part Fully Distributed setting [6], in which the dataset is distributed over a large number of users, each record is owned by two different users, and one user only knows the value for a subset of the attributes while the other knows the values for the remaining attributes. Miner aims to build an ID3 decision tree while protecting the privacy of each user.

Although there have been numerous studies on the privacy-preserving ID3 Algorithm, these studies are limited to two-party horizontal partitioning data mode [7], or horizontal partitioning data model with more than two-party [8, 9, 10, 11, 12, 13, 14, 15], or vertical partitioning data model with more than two-party [16, 17, 18, 14]. Therefore, they cannot be applied to the 2PFD setting.

Our problem can be solved by using the available solutions such as [14, 19]. However, due to the characteristics of the 2PFD setting, letting the parties exchange directly and sequentially with each other like the above solutions will lead to large communication costs and time costs. Furthermore, these solutions also assume that each pair of participants has a separate channel.

In this paper, we develop a privacy-preserving ID3 decision tree solution in the 2PFD setting. This solution does not require communication channels between different users. Additionally, many phases can be performed in parallel. First, we rewrite the formula that determines the best attribute. Then, we use the privacy-preserving frequency computation protocol in the 2PFD setting [20] to develop the privacy-preserving entropy of attribute protocol. Using this protocol, we construct the privacy-preserving ID3 decision tree solution. Finally, we evaluate the solution's performance and privacy.

The remainder of the paper is structured as follows: Section 2 reviews some technical preliminaries used in this work. Our protocol is described in Section 3. Finally, we will be the conclusion of the paper.

II. PRELIMINARIES

A. ELLIPTIC CURVE CRYPTOGRAPHY

Elliptic curve cryptography (ECC) is a public-key cryptosystem based on the discrete logarithm problem of elliptic curves over finite fields. ECC is well-known for its smaller key size and faster for the same level of security than other public-key cryptosystems (like RSA) [21].

Let E(F_p) be an Elliptic curve over a finite field F_p with a point O at infinity and p be a large prime, in which elliptic curve discrete logarithm problem is hard. In addition, G is a base point of the elliptic curve E with order q (i.e., $q.G = O$). The private key is the random number $d \in [1, q-1]$, and the corresponding public key curve point is $Q = d.G$. To encrypt the plaintext m, the sender uses the public key Q to compute the ciphertext C from the plaintext m as follows: he randomly chooses k from $[1, q-1]$ and computes the ciphertext $C = (C_1 = P_m + k.Q, C_2 = k.G)$ where P_m is a point of E with $x_{P_m} = m$. To decrypt the ciphertext C using the private key d, the receiver may compute $m = x_M$, in which $M = C_1 + (-d.C_2)$.

Under the decisional Diffie-Hellman assumption [22] for the curve E, the elliptic curve analog of the ElGamal system is semantically secure.

B. THE ID3 ALGORITHM

The main purpose of the algorithm is to construct a decision tree from a data set of examples and their classes using information theory. The ID3 algorithm builds a decision tree in a top-down manner with information about the patterns.

The best object classification will be obtained by starting at the root. The information gain is used to compute the best prediction. An attribute A_i's information gain is defined [9] as

$$Gain(S, A_i) = Entropy(S) - Entropy(A_i)$$

where, $Entropy(S)$: the entropy of a data set of tuples S (p is the total number of different values the target class can take on S), is defined as:

$$Entropy(S) = \sum_{i=1}^{p}\left(-\frac{|S_{c^i}|}{|S|}log_2\frac{|S_{c^i}|}{|S|}\right)$$

with $|S|$ and $|Si|$ are the number of tuples in S and the number of tuples in S having value c^i for the class attribute, respectively.

$Entropy(A_i)$: the entropy of A_i attribute, is defined as:

$$Entropy(A_i) = \sum_{j=1}^{e_i} \frac{|S_{a_i^j}|}{|S|} Entropy(S_{a_i^j})$$

where e_i is the number of possible values for the attribute A_i.

$S_{a_i^j}$ the subset of S with tuples having value a_i^j for attribute A_i.

In ID3, at each node, the selected attribute is determined based on:

$$A^* = \underset{A_i}{\operatorname{argmax}} Gain(S, A_i)$$

$$= \underset{A_i}{\operatorname{argmin}}\ Entropy(A_i)$$

i.e. the attribute that makes the information gain maximum.

The ID3 algorithm is shown in Figure 1.

Input: A, a set of attributes.

C, the class attribute.

S, data set of tuples.

Output: Decision tree

1: **if** A is empty **then**

2: **Return** the leaf having the most frequent value in S.

3: **else if** all tuples in S have the same class value **then**

4: **Return** a leaf with that specific class value.

5: **else**

6: Determine attribute A_i with the highest information gain in S.

7: Create a node that is not a leaf node A_i.

8: **for** $(1 \le j \le e_i)$ // *e_i is the number of values of attribute A_i.*

9: $ID3(A - \{A_i\}, C, S(a_i^j))$, with a_i^j, are the different values of A_i.

10: **End**

11: **Return** a tree with root A_i and e_i branches labeled a_i^1, ..., $a_i^{e_i}$, such that branch j contains $ID3(A - \{A_i\}, C, S(a_i^j))$.

12: **end if**

Figure 1. The ID3 Algorithm

C. The privacy-preserving frequency computation protocol in 2PFD setting

In this section, we briefly introduce the privacy-preserving frequency computation protocol in the 2PFD setting is proposed in [20] as follows:

Let E(Z_d) be an elliptic curve with a point O at infinity and d be a large prime, in which the elliptic curve discrete logarithm problem is hard. In addition, G is a base point of the elliptic curve E with order d (i.e., $d.G = O$).

Each user U_i keeps a private value $u_i \in \{0,1\}$. Nobody knows this value, beyond him. Before the PPFM protocol starts, each user chooses three private keys $x_i, y_i, z_i \in [1, d-1]$, after that he computes the corresponding public keys $X_i = x_i.G, Y_i = y_i.G, Z_i = z_i.G$. These public keys are sent to the miner before the protocol starts.

Each user V_i keeps a private value $v_i \in \{0,1\}$. Nobody knows this value, beyond him. Before the PPFM protocol starts, each user chooses three private keys $p_i, q_i, s_i \in [1, d-1]$, after that he computes
the corresponding public keys $P_i = p_i.G, Q_i = q_i.G, S_i = s_i.G$. These public keys are sent to the miner before the protocol starts.

The privacy-preserving frequency co protocol in 2-PFD consists of five phases described in Fig. 2.

Phase 1 :

- Each user U_i

Choose three private keys $x_i, y_i, z_i \in [1, d-1]$, Compute $X_i = x_i.G, Y_i = y_i.G, Z_i = z_i.G$

Send $X_i||, Y_i$ to **Miner**

- Each user V_i

Choose three private keys $p_i, q_i, s_i \in [1, d-1]$, Compute $P_i = p_i.G, Q_i = q_i.G, S_i = s_i.G$

Send $P_i||, Q_i$ to **Miner**

- Miner:

Compute:

$X = \sum_{i=1}^{n}(X_i + P_i) = x.G,\ \ Y = \sum_{i=1}^{n}(Y_i + Q_i) = y.G$

Phase 2: Each user U_i

Choose a random number $c_i \in [0, d-1]$,

Compute $C_1^{(i)} = u_i.G + c_i.Z_i$ và $C_2^{(i)} = c_i.G$

Send $C_1^{(i)}||\ C_2^{(i)}$ to **Miner**

Phase 3: Each user V_i

Get $C_1^{(i)}||\ C_2^{(i)}$ from **Miner**

Choose a random number $r_i \in [0, d-1]$,

Compute $R_1^{(i)} = v_i.C_1^{(i)} + q_i.X$, $R_2^{(i)} = s_i.r_i.C_2^{(i)} + p_i.Y$, $R_3^{(i)} = r_i.S_i - v_i.Z_i$

Send $R_1^{(i)}||\ R_2^{(i)}||\ R_3^{(i)}$ to **Miner**

Phase 4:

Get $R_1^{(i)}||\ R_2^{(i)}||\ R_3^{(i)}$ from **Miner**

Compute $M_i = R_1^{(i)} + c_i.R_3^{(i)} - R_2^{(i)} - x_i.Y + y_i.X$

Send M_i to **Miner**

Phase 5: Miner computes:

$$M = \sum_{i=1}^{n} M_i$$

Find the satisfying trị f value: M=$f.G$ using the brute force algorithm.

Figure 2. The privacy-preserving frequency computation protocol in the 2PFD setting

III. PRIVACY-PRESERVING DECISION TREE SOLUTION

In this section, we will discuss a privacy-preserving ID3 decision tree solution in the 2PFD setting. Furthermore, the miner only knows what attributes are in the system and their respective value domains but not who owns them.

A. PROBLEM STATEMENT

We consider the 2PFD setting: There are m attributes, $A_1, A_2, \dots, A_k, \dots, A_m$ and one class attribute C. and one class attribute A_i $(1 \le i \le m)$ can take the values $a_i^1, a_i^2, \dots, a_i^{e_i}$, and C can take the values $c^1, c^2, \dots, c^p$. Assume that there are $2n$ users $\{U_1, U_2, \dots, U_n\}$ and $\{V_1, V_2, \dots, V_n\}$. Each pair $(U_l,\ V_l)$ owns a vector $(a_{l,1}, a_{l,2}, \dots, a_{l,m}, c_l)$, where $(a_{l,1}, a_{l,2}, \dots, a_{l,k})$ denote an instance of the attribute vector $(A_1, A_2, \dots, A_k)$ that owned by U_l, and $(a_{l,k+1}, \dots, a_{l,m}, c_l)$ denote an instance of the attribute vector $(A_{k+1}, A_2, \dots, A_m)$ and its class label owned by V_l as illustrated in Figure 3. Our purpose is to allow the miner to train the decision tree using data from all users while protecting the privacy of each user.

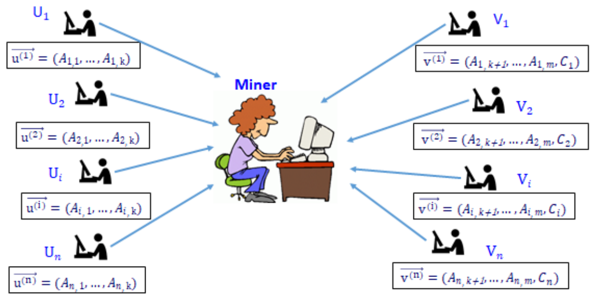

Figure 3. The data model

B. SECURE ATTRIBUTE ENTROPY PROTOCOL

We use frequency mining as an initial method to design a privacy-preserving attribute entropy protocol for building ID3 decision trees. The two requirements of the protocol are accuracy and privacy, where the protocol allows the miner to correctly construct an ID3 decision tree without knowing anything about each user's sensitive data. We rewrite the formula to choose the best attribute as follows:

$$A^* = \underset{A_i}{\operatorname{argmin}}\ Entropy(A_i)$$
$$= \underset{A_i}{\operatorname{argmin}} \sum_{j=1}^{e_i} \frac{|S_{a_i^j}|}{|S|} Entropy(S_{a_i^j})$$
$$= \underset{A_i}{\operatorname{argmin}} \sum_{j=1}^{e_i} (\frac{f(a_i^j,*,S)}{f(*,*,S)}.$$
$$\sum_{k=1}^{p} (-\frac{f(a_i^j,c^k,S)}{f(a_i^j,*,S)} log_2 \frac{f(a_i^j,c^k,S)}{f(a_i^j,*,S)})$$

where, $f(*,* S)$ is the number of tuples in S, $f(a_i^j,*,S)$ is the frequency of $(A_i = a_i^j)$ in S $(f(*,*,S) = \sum_{j=1}^{e_i} f(a_i^j,*,S) = \sum_{j=1}^{e_i} f(*,a_i^j,S)$), $f(a_i^j,c^k,S)$ is the frequency of the pair $(A_i = a_i^j, c = c^k)$ in S $(f(a_i^j,*,S) = \sum_{k=1}^{p} f(a_i^j,c^k,S))$.

Thus, the computing $Entropy(A_i)$ based on the frequencies $f(a_i^j,c^k,S)$ in S $(1 \le i \le m, 1 \le k \le p, 1 \le j \le e_i)$.

Therefore, our protocol allows the miner to obtain the attribute entropy by privately computing the frequencies $f(a_i^j,c^k,S)$ by using the primitive presented in Section II.D. This protocol does not reveal any of each user's privacy information to the miner beyond the frequencies in all user's data. Furthermore, the protocol keeps the miner in the dark about the set of attributes that each user has. For more convenience, in the proposed protocol, we denote T_{ij} be a tuple of the domain $D_i \times C$. Here D_i is the domain of the attribute A_i, C is the domain of the attribute, j $(j = 1,..,t)$ is the index of the j^{th} tuple in the domain $D_i \times C$, $t = |D_i \times C|$, and the first value and the second value of the tuple T_{ij} are denoted by $T_{ij}.a$ and $T_{ij}.c$, respectively.

We assume that each user has private keys and public keys as presented in Section II.D. Note that the security of ciphertext depends on new random values being used for each encryption. In the frequency mining protocol, the x_l, y_l, p_l and q_l are random values, and associated X and Y cannot be reused in different uses of the protocol. Therefore, in the protocol of privacy-preserving attribute entropy, with each computed frequency, each U_l chooses a random element in Z_d to randomize its public keys that results in the randomization of parameters X, Y in each done computation. In particular, if the protocol is to be run many times, many randomizations of values X and Y could be implemented so that keys x_l, y_l, p_l and q_l can be reused. Our protocol is depicted as follow:

• **Phase 1:** Each user U_l work as follows. For $1 \le i \le m, 1 \le j \le t$

- If U_l owns A_i then

  If $a_{li} = T_{ij}.a$ then $u_{l,1} = else\ u_{l,1} = 0$

  Else

  $u_{l,1} = 1$.

- If U_l owns C then

  If $c_l = T_{ij}.c$ then $u_{l,2} = 1$ else $u_{l,2} = 0$

  Else

  $u_{l,2} = 1$.

- If $T_{ij} \in S$ then $u_{l,3} = 1$ else $u_{l,3} = 0$.

- Compute $u_l = u_{l,1}.u_{l,2}.u_{l,3}$.

- Randomly choose k_l and b_l from $\{1,\dots,d-1\}$, compute $C_{ij}^{l1} = u_l.G + b_l.Z_l$, $C_{ij}^{l2} = b_l.G$, $C_{ij}^{l3} = k_l.X_l$, and $C_{ij}^{l4} = k_l.Y_l$,

- Send C_{ij}^{l1}, C_{ij}^{l2}, C_{ij}^{l3} và C_{ij}^{l4} to Miner,

- The Miner computes:

$$X_{ij} = \sum_{l=1}^{n} (C_{ij}^{l3} + P_l)$$

and

$$Y_{ij} = \sum_{l=1}^{n} (C_{ij}^{l4} + Q_l)$$

• **Phase 2:** Each user V_l works as follows. For $1 \le i \le m, 1 \le j \le t$

- If V_l owns A_i then

  If $a_{li} = T_{ij}.a$ then $v_{l,1} = else\ v_{l,1} = 0$

  Else

  $v_{l,1} = 1$.

- If V_l owns C then

  If $c_l = T_{ij}.c$ then $v_{l,2} = 1$ else $v_{l,2} = 0$

Else

$v_{l,2} = 1.$

- If $T_{ij} \in S$ then $v_{l,3} = 1$ else $v_{l,3} = 0$.

- Compute $v_l = v_{l,1}.v_{l,2}.v_{l,3}$,

- Get C_{ij}^{l1}, C_{ij}^{l2}, X_{ij} và Y_{ij} from Miner,

- Randomly choose r_l from $\{1,\dots,d-1\}$,

- Compute $R_{ij}^{l1} = v_l.C_{ij}^{l1} + q_l.X_{ij}$, $R_{ij}^{l2} = s_l.r_l.C_{ij}^{l2} + p_l.Y_{ij}$ and $R_{ij}^{l3} = r_l.S_l - v_l.Z_l$,

- Send R_{ij}^{l1}, R_{ij}^{l2} and R_{ij}^{l3} to Miner.

• **Phase 3:** Each user U_l works as follows. For $1 \le i \le m, 1 \le j \le t$

- Get R_{ij}^{l1}, R_{ij}^{l2}, R_{ij}^{l3}, X_{ij} and Y_{ij} from Miner,

- Compute $M_{ij} = R_{ij}^{l1} + b_l.R_{ij}^{l3} - R_{ij}^{l2} - k_l x_l.Y_{ij} + k_l y_l.X_{ij}$ and send it to Miner.

• **Phase 4:** The Miner works as follows.

- For $1 \le i \le m, 1 \le j \le t$

* Compute $d_{ij} = \sum_{l=1}^{n} M_{ij}$

* Find $f(a_i^j, c^k, S)$ that satisfies: $f(a_i^j, c^k, S)\ .G = d_{ij}$ using the brute force algorithm, for $1 \le i \le m, 1 \le j \le e_i, 1 \le k \le p$.

- Compute $f\left(a_i^j,*,S\right) = \sum_{k=1}^{p} f(a_i^j, c^k, S)$.

- Compute $f(*,*,S) = \sum_{j=1}^{e_i} f(a_i^j,*,S)$.

- Miner outputs Entropy of A_i in S: $Entropy(A_i) = \sum_{j=1}^{e_i}\left(\frac{f(a_i^j,*,S)}{f(*,*,S)}.\sum_{k=1}^{p}\left(-\frac{f(a_i^j,c^k,S)}{f(a_i^j,*,S)} log_2 \frac{f(a_i^j,c^k,S)}{f(a_i^j,*,S)}\right)\right).$

Basically, the correctness and privacy of our privacy-preserving attribute entropy protocol can be derived from the frequency computing in Section II.C.

Theorem 3.1. The protocol presented in Section II.B allows the miner to obtain attribute entropy correctly.

Proof. By the protocol [20] correctly computes each $f(a_i^j, c^k, S)$. correctly computes each $f\left(a_i^j,*,S\right), f(*,*,S)$ can be directly obtained from frequencies $f(a_i^k, c^j, S)$ by the formula:

$$f\left(a_i^j,*,S\right) = \sum_{k=1}^{p} f(a_i^j, c^k, S)$$

$$f(*,*,S) = \sum_{j=1}^{e_i} f(a_i^j,*,S)$$

Therefore, the protocol outputs attribute entropy correctly.

Định lý 3.2. This protocol preserves the privacy of the honest users against the miner and up to $2n - 2$ corrupted users. In cases with only two honest users, it remains correct as long as two honest users do not own the attribute values of the same record.

Proof. Note that in the protocol, the values k_l, b_l and r_l are independently and randomly chosen for every frequency value, so the computation is independently done for every frequency, therefore this corollary follows immediately from the privacy-preserving frequency computing protocol in [20].

From the above two theorems, this protocol ensures accuracy and privacy.

C. SECURE ID3 DECISION TREE ALGORITHM

It is assumed that each user's data includes sensitive attribute values (without loss of generality, assuming that all attribute values of each user are sensitive). As a result, no user is prepared to give the miner his data without protecting privacy. Furthermore, the miner does not know what attributes the user owns, but only knows the set of attributes and their value domain. To allow the miner to build a decision tree while protecting the privacy of each user, we design a privacy-preserving decision tree solution.

The miner implements the ID3 decision tree algorithm as follows:

Input: A, a set of attributes.
C, the class attribute.
S, data set of tuples.
Output: Decision tree
1: **if** $(A = \emptyset)$ **then**

2: **Return** $c^* = \underset{c^k}{\text{argmax}}\, f(*, c^k, S)$

3: **else if** $(f(*, c^k, S) = f(*,*,S))$ **then**

4: **Return** c^k.

5: **else**

6: Execute *the privacy-preserving attribute entropy protocol* for all attributes $A_i \in A$

7: Choose the attribute A_i with the highest information gain in S as the node.

8: **for** $(1 \leq j \leq e_i)$

9: $ID3(A - \{A_i\}, C, S(a_i^j))$, with a_i^j are different values of A_i.

10: **End**

11: **Return** a tree with root A_i and e_i branches labeled a_i^1, ..., $a_i^{e_i}$, such that branch j contains $ID3(A - \{A_i\}, C, S(a_i^j))$.

12: **end if**

Figure 4. The privacy-preserving decision tree in the 2PFD setting

D. SOLUTION EVALUATION

We assess the proposed solution's correctness, privacy, and performance.

1. Correctness analysis

The security frequency computation protocol in Section II.C and the secure attribute entropy computation protocol in Section III.B can be used to determine the correctness of the privacy-preserving ID3 decision tree solution.

Corollary 3.1. *The proposed solution allows the miner to get the correct ID3 decision tree.*

Proof. The secure attribute entropy protocol in Section III.B correctly computes each $A_i \in A$ in S.

The privacy-preserving frequency computing protocol in [20] correctly computes each $f(*, c^k, S)$. Furthermore, $f(*,*,S)$ can be directly obtained from frequencies $f(*, c^k, S)$ by the formula:

$$f(*,*,S) = \sum_{j=1}^{e_i} f(*, c^k, S)$$

Thus, the protocol outputs the ID3 decision tree correctly.

2. Privacy analysis

The privacy-preserving frequency computing protocol in section II.C and the secure attribute entropy computing protocol in section III.B, respectively, can be used to provide privacy in this solution.

Corollary 3.2. *The proposed solution preserves the privacy of the honest users against the miner and up to 2n - 2 corrupted users. In cases with only two honest users, it remains correct as long as two honest users do not own the attribute values of the same record.*

Proof. Another key theory that we adopt to prove the privacy preservation property of the proposed solution is the Composition Theorem under the semi-honest model (Theorem 3.3). Detailed proof of Theorem 3 could be found in [11], and thus is omitted here.

Theorem 3.3 (Composition theorem for the semi-honest model, multi-party case) [11]. *Suppose that the m-ary functionality g is privately reducible to the k-ary functionality f and that there exists a k-party protocol for privately computing f. Then there exists an m-party protocol for privately computing g.*

According to this theorem, in the semi-honest model, if a protocol is built on the concatenation of many (proven) secure subprotocols, then the protocol is also secure. Thus combined with the computation being performed independently for all frequencies, this consequence follows right from the privacy-preserving frequency computing protocol and the secure attribute entropy computing protocol.

3. Communication and Computational cost

Next, we compare the performance of our solution with the solution in [14]. We'll refer to denote m as the number of non-class attributes, p as the number of class attribute valuend k as the maximum number of non-attribute values class, t is the length of the encryption key (t is usually very large).

In our solution, to determine the best data classifier attribute, m secure attribute entropy computing protocols need to be implemented. In each of these protocols, each user U_i needs to compute $5pk$ ciphertext in phase 1 and phase 3, each user V_i computes $3pk$ ciphertext in phase 2, miner computes $2pk$ ciphertext sum of $2n$ ciphertext in phase 1, and pk ciphertext sum of n ciphertexts in phase 4, since in phases users U_i and V_i are assumed to execute concurrently, computation cost $= O(m(8 + 5n)pkt)$. In terms of communication costs, each user U_i sends $4pk$ messages to the miner in phase 1, receives $5pk$ messages from the miner, and sends pk messages to the miner in phase 3. Each user V_i sends 2 messages to the miner in phase 1, receives $4pk$ messages from the miner, and sends $3pk$ messages to the miner in phase 3, so communication cost $= O(m(2 + 17pk)nt$. The grid will be 2 horizontally and n vertically in the solution of horizontal merge and vertical development [14], therefore the computation cost is $O(m(n + k + p)4nt^3)$ and the communication cost is $O(m(n + p)4nt)$. As a result, the proposed procedure is more efficient than [14].

IV. CONCLUSION

In this paper, we have proposed a privacy-preserving ID3 decision solution in the 2PFD setting. This solution allows the miner to correctly construct the ID3 decision tree while maintaining the privacy of each user's sensitive data in the 2PFD setting. It even ensures the privacy of the user's attribute ownership model.

We will continue to research privacy-preserving data mining solutions in the 2PFD setting model in the future.

REFERENCES

[1] Agrawal R., Srikant R, "Privacy-Preserving Data Mining," in The ACM SIGMOD Conference, 2000.

[2] Vassilios S. Verykios, Elisa Bertino, Igor Nai Fovino, Loredana Parasiliti Provenza, Yucel Saygin, Yannis Theodoridis, "State-of-the-art in Privacy Preserving Data Mining," SIGMOD Rec., vol. 33, no. 1, 2004.

[3] S. Mehnaz, and G. Bellala and E. Bertino, "A secure sum protocol and its application to privacy-preserving multi-party analytics," in Proc. of the 22nd ACM on Symposium on Access Control Models and Technologies, 2017.

[4] Supriya, "A Survey on Privacy Preserving Data Mining Techniques," International Journal of Emerging Technology and Advanced Engineering, pp. 119-122, 2015.

[5] R. Mendes and J. P. Vilela, "Privacy-preserving data mining: methods," in IEEE Access, 2017.

[6] The Dung Luong, Tu Bao Ho, "Privacy Preserving Frequency Mining in 2-Part Fully Distributed," IEICE Transactions on Information and Systems, pp. 2702-2708, 2010.

[7] Y. Lindell and B. Pinkas, "Privacy preserving data mining," in Annual International Cryptology Conference, 2000.

[8] "Privacy preserving ID3 algorithm over horizontally partitioned data," in Parallel and Distributed Computing, Applications and Technologies, 2005. PDCAT 2005. Sixth International Conference on, 2005.

[9] Saeed Samet and Ali Miri, "Privacy preserving ID3 using Gini index over horizontally partitioned data," in Computer Systems and Applications, 2008. AICCSA 2008. IEEE/ACS International Conference on, 2008.

[10] de Hoogh S., Schoenmakers B., Chen P., op den Akker H., " Practical Secure Decision Tree Learning in a Teletreatment Application," in Financial Cryptography and Data Security. FC 2014. Lecture Notes in Computer Science, vol 8437, Springer, Berlin, Heidelberg, 2014.

[11] Ye Li, Xuan Wang, Zoe L. Jiang, S.M. Yiu, "Outsourcing privacy-preserving ID3 decision tree over horizontally partitioned data for multiple parties," International Journal of High Performance Computing and Networking (IJHPCN), vol. 12, no. 1,
pp. 207-215, 2018.

[12] Ye Li, Zoe L. Jiang , Xuan Wang , Junbin Fang, En Zhang , and Xianmin Wang, "Securely Outsourcing ID3 Decision Tree in Cloud Computing," Wireless Communications and Mobile Computing, vol. 2018, pp. Article ID 2385150, 10 pages, 2018.

[13] Stacey Truex, Nathalie Baracaldo, Ali Anwar, Thomas Steinke. Heiko Ludwig and Rui Zhang, "A Hybrid Approach to Privacy-Preserving Federated

Learning," CoRR, vol. abs/1812.03224, p. 11 pages, 2018.

[14] Bart Kuijpers, Vanessa Lemmens, Bart Moelans and Karl Tuyls, "Privacy Preserving ID3 over Horizontally, Vertically and Grid Partitioned Data," CoRR, vol. abs/0803.1555, 2008.

[15] Kamini D. Tandel and Jignasa N. Patel, "Privacy Preserving Decision Tree Classification on Horizontal Partition Data," in International Journal of Engineering Research & Technology (IJERT), 2016.

[16] Justin Zhan, Stan Matwin, and LiWu Chang, "Privacy-Preserving Decision Tree Classification," in Proceedings of the Fifth International Conference on Electronic Business, 2005, pp. 470 - 476.

[17] Vaidya, Jaideep and Clifton, Chris and Kantarcioglu, Murat and Patterson, A. Scott, "Privacy-Preserving Decision Trees over Vertically Partitioned Data," ACM Trans. Knowl. Discov. Data, vol. 2, no. 3, p. 1–27, 2008.

[18] Fang, Weiwei and Yang, Bingru, "Privacy Preserving Decision Tree Learning over Vertically Partitioned Data," in 2008 International Conference on Computer Science and Software Engineering, 2008, pp. 1049-1052.

[19] Shuguo HAN, and Wee Keong NG, "Multi-Party Privacy-Preserving Decision Trees for Arbitrarily Partitioned Data," INTERNATIONAL JOURNAL OF INTELLIGENT CONTROL AND SYSTEMS, vol. 2, no. 4, pp. 351-358, 2007.

[20] Thi Van Vu, The Dung Luong, Van Quan Hoang, "An Elliptic Curve-based Protocol for Privacy Preserving Frequency Computation in 2-Part Fully Distributed Setting," in 12th International Conference on Knowledge and Systems Engineering (KSE), Can Tho, 2020.

[21] Christof Paar and Jan Pelzl, Understanding Cryptography, Springer, Berlin, Heidelberg, 2010.

[22] Canetti, Ran, "Decisional Diffie-Hellman Assumption," in Encyclopedia of Cryptography and Security, Boston, MA, Springer US, 2005, pp. 140-142.

[23] Steven D. Galbraith, Ping Wang and Fangguo Zhang, "Computing Elliptic Curve Discrete Logarithms with Improved," Cryptology ePrint Archive, 2015.

[24] Siraj, Maheyzah Md and Rahmat, Nurul Adibah and Din, Mazura Mat, "A Survey on Privacy Preserving Data Mining Approaches and Techniques," in Proceedings of the 2019 8th International Conference on Software and Computer Applications, New York, NY, USA, Association for Computing Machinery, 2019, p. 65–69.

[25] Chris Clifton, Murat Kantarcioglu,Jaideep Vaidya, Xiaodong Lin, Michael Y. Zhu, "Tools for Privacy Preserving Distributed Data Mining," SIGKDD Explorations, vol. 4, no. 2, pp. 1-5, 2002.

[26] Babak Siabi, Mehdi Berenjkoub, Willy Susilo, "Optimally Efficient Secure Scalar Product With Applications in Cloud Computing," IEEE Access, vol. 7, no. 1, pp. 42798 - 42815, 29 3 2019.

[27] C. Dong, L. Chen, "A Fast Secure Dot Product Protocol with Application to Privacy Preserving Association Rule Mining," in Proc. of Pacific-Asia Conference on Knowledge Discovery and Data Mining, 2014.

[28] Y. Zhu,T. Takagi, "Efficient scalar product protocol and its privacypreserving application," in International J. of Electronic Security and Digital Forensics, Vol. 7, No. 1, 2015.

[29] Goethals, Bart and Laur, Sven and Lipmaa, Helger and Mielikainen, Taneli, "On private scalar product computation for privacy-preserving data mining," in Proc. of International Conference on Information Security and Cryptology, 2004.

[30] V.Duy-Hien, L.The-Dung, H.Tu-Bao, "An Efficient Approach for Secure Multiparty Computation without Authenticated Channel," in Information Science, 2019.

ABOUT THE AUTHOR

V.

Vu Thi Van

Workplace: Academy of Cryptography Techniques

Email: vanvu10101986@gmail.com

Education: Engineer of Information Security from the Academy of Cryptography, Hanoi, Viet Nam, in 2009, Master's degree in Information Security from Academy of Cryptography Techniques, in 2016. She is currently a Ph.D. candidate of Infomation security, Academy of Cryptography, Vietnam.

Recent research direction: secure mutli-party computation, data mining, cyber security, etc.

Luong The Dung

Workplace: Academy of Cryptography Techniques

Email: thedungluong1@gmail.com

Education: Received Bachelor of Information Technology from Le Quy Don Technical University, Ha Noi, Viet Nam in 2001 and a Ph.D. degree in 2012 from Institute of Military Science and Technology, Ha Noi, Viet Nam.

Recent research direction: privacy-preserving data mining and computer security, etc.

Hoang Van Quan

Workplace: Staff General, Ministry of Defense

Email: hoangvanquan@gmail.com

Education: Engineer of Cryptographic Technique of The Academy of Cryptography, Hanoi, Viet Nam, in 1994; Master's degree in Information and Electronic Engineering from Military Institute of Science and Technology, in 2005; Ph.D. of Electronic Engineering at Military Institute of Science and Technology in 2016.

Recent research direction: Cryptography.

Tran Thi Luong

Workplace:Academy of Cryptography Techniques, No.141 Chien Thang road, Tan Trieu, Thanh Tri, Hanoi, Vietnam

Email: luongtran@actvn.edu.vn

Education: Bachelor of Mathematics and Informatics of The Ha Noi university of Sicence in 2006; Master degreee in cryptographic technique at Academy of Cryptographic Techniques in 2012; Phd degree in cryptographic technique at Academy of Cryptographic Techniques in 2019;

Recent research direction: Cryptogaphy, Coding theory and Information Security.

Hoang Duc Tho

Workplace: Academy of Cryptography Techniques.

Email: thohd80@gmail.com

Education: Philosophy of Doctor.

Recent research direction: cryptography, information security.