Wireless Networks (2022) 28:303–317
https://doi.org/10.1007/s11276-021-02846-1

ORIGINAL PAPER

Privacy-preserving association rule mining based on electronic medical system

Wenju Xu[1] · Qingqing Zhao[1] · Yu Zhan[1] · Baocang Wang[1,2] · Yupu Hu[1]

Accepted: 11 November 2021 / Published online: 3 January 2022

Abstract
Privacy protection during collaborative distributed association rule mining is an important research, which has been widely used in market prediction, medical research and other fields. In medical research, Domadiya et al. (Sadhana 43(8):127, 2018) focused on mining association rules from horizontally distributed healthcare data to diagnose heart disease. They claimed they proposed a more effective privacy-preserving distributed association rule mining (PPDARM) scheme. However, a serious security scrutiny of the scheme is performed, and we find it vulnerable to protect the support of the itemsets from any electronic health record (EHR) system, which is the most important parameter Domadiya et al. tried to protect. In this paper, we first present the cryptanalysis of the PPDARM scheme proposed by Domadiya et al. as well as some revised performance analyses. Then a new PPDARM scheme with less interactions is proposed to avert the shortcomings of Domadiya et al., using the homomorphic properties of the distributed Paillier cryptosystem to accomplish the cooperative computation. Our scheme allows the directed authority (miner) to obtain the final results rather than all cooperative EHR systems, in case of semi-honest but pseudo EHR systems. Moreover, security analysis and performance evaluation demonstrate our proposal is efficient and feasible.

Keywords Privacy-preserving · Association rule mining · Homomorphic encryption · Cooperative computation

1 Introduction

In the electronic era, big data has attracted increasing attention from various trades and industries, especially with the rapid development of wireless networks [1] and big data. Since the datum contain a variety of information, any individuals or organizations can make full use of data to mine the external or potential information for unpredictable profits. Then the research of data mining [2] is meaningful. In fact, data mining, also known as database knowledge discovery, is a useful tool and can be applied to discover the relationship among a large number of random data sets in market prediction, medical research and other fields, which is generally divided into four categories: association rule mining [3], classification mining [4], cluster mining [5], and prediction mining [6].

As one of the main branches of data mining, association rule mining and its fundamental step frequent itemset mining [7] are two popularly and widely studied data analysis techniques for a range of market prediction and medical research [8]. With the mining result, the supermarket can arrange their goods appropriately which can help gain more profits, meanwhile provide a big convenience for their customers. Similarly, the medical staff, such as the emergency medical technicians, can predict the disease of the patients from the symptoms and causes, then take some effective medical treatments under any critical

✉ Baocang Wang
bcwang@xidian.edu.cn

Wenju Xu
xuwenjuxwj@126.com

Qingqing Zhao
1093806982@qq.com

Yu Zhan
yzhan1993@163.com

Yupu Hu
yphu@mail.xidian.edu.cn

[1] State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an 710071, People's Republic of China

[2] Cryptographic Research Center, Xidian University, Xi'an 710071, People's Republic of China

situations to preliminarily save a life from the death. However, the data especially the medical data sets include many sensitive information of persons, such as gender, age, address, career, ID number, medical history, various indexes of medical examination and so on, so privacy protection on association rule mining of medical research [9–12] is pretty popular.

The Apriori algorithm proposed by Agarwal et al. [10] for association rule mining has received a greater attention. It first generated all the local frequent itemsets at each node and then communicated to the general node, producing the global frequent itemset of association rules. Nahar et al. [11] investigated the association rule mining to detect the sick and healthy factors which contributes to heart disease for males and females, along with the three rule generation algorithms: Apriori, Predictive Apriori and Tertius. In 2016, Qamara et al. [13] addressed the relation between the medical datasets and clinically-relevant patterns without endangering the privacy of patients. A hybrid privacy-preserving clinical decision support system from medical data in fog-cloud computing was proposed by Liu et al. [14]. Baroni et al. [15] proposed a novel divergent association rules approach to obtain thousands of association rules from the datasets of the malaria in Brazil.

Considering the digitization of e-health, the electronic health record (EHR) system [16] can not only store large amounts of medical data but also perform some computations as carte. Jensen et al. [17] utilized EHR data for further medical research and clinical care. Then the data collected by the EHR system can be used for privacy-preserving association rule mining of medical research and disease diagnosis. Gkoulalas et al. [18] presented a survey of algorithms for the patient data in EHR systems, which remains useful for subsequent analysis tasks in a privacy-preserving way. Privacy-preserving association rule mining for horizontally partitioned healthcare data [19] and vertically partitioned healthcare data [20] were both proposed by Domadiya and Rao respectively. Yigzaw et al. [21] proposed a feasible architecture which can protect the privacy of the patients, clinicians, and healthcare institutions as well as mine the clinical performance of a clinician over the patient data from EHR systems.

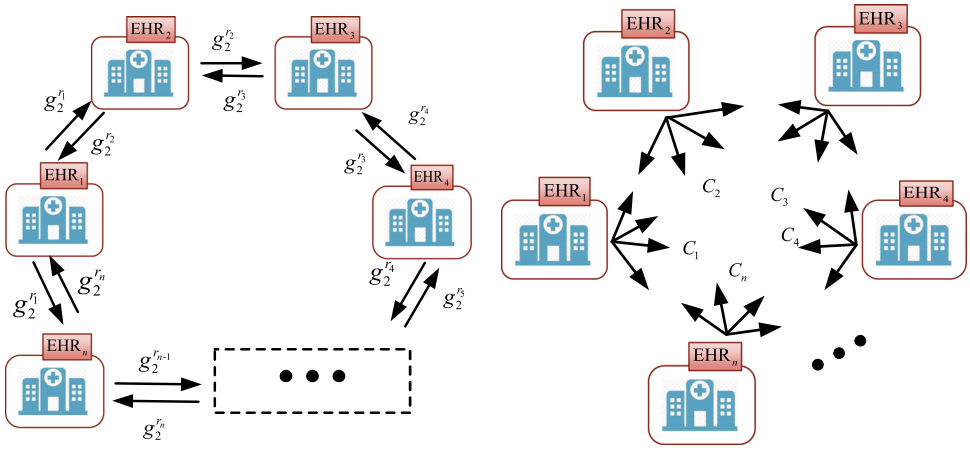
Motivation Faced with the rampant Covid-19 virus under the national circumstance, association rule mining of medical data becomes urgent and meaningful, since it can provide some auxiliary references for the research of vaccine. For example, the Health Committee wants to know the situation of the antibody after the first dose of the vaccine from volunteers during the research and development of vaccine. Statistically speaking, collecting more information of the antibody for volunteers from various locations is preferred. Hence, we aim to help the Health Committee to attain the situation of the antibody for volunteers from various locations without leaking the information of the volunteers, in order to investigate the availability of vaccine. Considering the above reality and requirements, we can first turn to a similar scenario for experiment and reference: a privacy-preserving distributed association rule mining (PPDARM) scheme for horizontally partitioned healthcare data.

Domadiya et al. [19] proposed a PPDARM scheme for horizontally partitioned healthcare data with insecure communication channel. They also claimed that the scheme can produce an accurate mining result without compromising the data privacy of every EHR system. The system model is shown as Fig. 1. More details can turn to Sect. 3 in this paper. However, we find the scheme has serious security flaws in protecting the data privacy of every EHR system. The improper points in [19] are summarized as follows.

- Every EHR system (participant) is required to make interactions with the two neighboring EHR systems. Although the threat model indicates the EHR systems are assumed to be honest but curious, the situation where the semi-honest (honest-but-curious) but pseudo EHR systems cannot be avoided. Moreover, the higher communication overhead is also an issue to be improved.
- The performance analyses of the PPDARM scheme in [19] are not exactly correct. Take the set-up phase for example. Every EHR system performs a division and a modular exponentiations modulo p^2 and then broadcasts the ciphertext to the neighboring two EHR systems. Note that the computation complexity of modular exponentiation and modular inverse is $O(\log^3 p)$,[1] the computation complexity for every EHR system is $O(\log^3 p)$. Although $\log^3 p$ can be regarded as a constant for security parameter of the scheme, the communication complexity is illogical to be $O(q)$ from $r_i \in \mathbb{Z}_q$ as Domadiya et al. claimed.
- The PPDARM scheme is not as secure as the security analysis claims under the computational discrete logarithm assumption to effectively protect the information of the supports. Instead of secure communication channel as [22], the honest-but-curious EHR systems can prevent active attacks including interrupt, tampering and forgery. However, we can still recover the support of the i-th EHR system $C_{count(i)}$ by computing $Y_i^q \bmod p^2$ due to the public parameter q, which is the order of the cyclic multiplicative group G_2, i.e., $R_i^q \equiv 1 \bmod p^2$.

[1] The symbol $O(\cdot)$ is commonly used asymptotic complexity notations. We denote an asymptotic upper bound with $O(\cdot)$.

Fig. 1 System model in [19]



(a) Set-up phase **(b)** Communication of encryption phase

Contribution In the following, we introduce an improved PPDARM scheme on healthcare data with insecure communication channel motivated by [19]. Note that the final goal of the PPDARM scheme in [19] is to ensure every semi-honest EHR system to obtain the sum of the supports, which cannot avoid pseudo EHR systems. Therefore, in our setting, only a directed authority, such as the Health Committee, is allowed to have access to the support sum of all the EHR systems.[2] Specifically, the contributions are unfolded below.

- Cryptanalysis of the PPDARM scheme in [19]. First, some errors and unclear description of the writings are illustrated. The theoretical performance of the PPDARM scheme is also evaluated again. Then we propose an attack algorithm to the PPDARM scheme to show it is not as secure as they claimed under the computational discrete logarithm assumption.
- A new PPDARM scheme based on electronic medical system. To avoid pseudo EHR systems to know the final results, we propose a PPDARM scheme which only allows the directed authority (miner) to obtain the final results rather than all EHR systems. In other words, only the miner is capable of disease prediction from all possible symptoms when preserving the privacy of all EHR systems.
- The security and performance of our proposal. The distributed Paillier cryptosystem is utilized to protect the supports of EHR systems. In addition, we perform the security analysis, theoretical analysis and experimental analysis to demonstrate the feasibility and availability of our PPDARM scheme based on electronic medical system.

[2] In our setting, we think the EHR systems may forge the support to obtain the final sum of the supports. Hence we make such an assumption in order to make the weakness in [19] not affect our scheme.

The structure of this paper is as follows. Section 2 introduces some notations, concepts and cryptosystems used throughout the paper. The PPDARM scheme in [19] is briefly described in Sect. 3, followed by our cryptanalysis. Our improved PPDARM scheme based on electronic medical system is elaborated in Sect. 4, including the problem description, model, specific scheme, security analysis and performance evaluations. Finally, we end the paper with conclusion in Sect. 5.

2 Preliminaries

We first introduce some basic notations in Table 1.

2.1 Frequent itemset mining and association rule mining

Association rule mining is an efficient way among different data mining methods, and frequent itemset mining is the fundamental step of association rule mining. Let $I = \{i_1, i_2, \ldots, i_m\}$ be the set of all the items, where i_j is an item for $j \in [m]$. In the electronic medical system, each item represents a symptom or a disease of the patient. The support of an itemset X described as $Supp(X)$ is the number of I in the EHR system that contains X. We get the mining result by conducting comparison between $Supp(X)$ and a support threshold $Supp_{min}$. Concretely, if $Supp(X) \geq Supp_{min}$, the mining result is that the itemset X is frequent; otherwise, X is infrequent.

Assume an association rule "$A \Rightarrow B$", it means that B will also occur under the premise that A occurs, where $A, B \subset I$ and $A \cap B = \emptyset$. The support and confidence of the rule "$A \Rightarrow B$" are two important indicators measuring the correlation of A and B. The support of A represents the frequency of A in the entire transaction data set,

Table 1 Notations

Notation	Description
n	Total number of EHR systems (participants)
$[n]$	$[n] = \{1, 2, \ldots, n\}$
mod$n$	$\{0, 1, 2, \ldots, n-1\}$
$\mathbb{Z}$	The integer ring
$\mathbb{Z}_n$	$\mathbb{Z}_n = \{0, 1, 2, \ldots, n-1\}$
gcd	The greatest common divisor
lcm	The lowest common multiple
$\mathbb{Z}_n^*$	$\mathbb{Z}_n^* = \{a \mid 0 < a < n, \gcd(a, n) = 1\}$
$\lvert n \rvert_2$	The length of the integer n
$m_i (i \in [n])$	The support of the i-th EHR system
$m_{\sum}$	Sum of the supports of all the EHR systems
$Supp_{min}$	Support threshold for frequent itemsets

$$Support(A) = P(A).$$

The confidence of the rule "$A \Rightarrow B$ " represents the proportion of the probability that A and B occur simultaneously in the probability of A occurring, that is

$$Confidence(A \Rightarrow B) = P(B \mid A) = \frac{P(A \cup B)}{P(A)}.$$

2.2 Cryptosystem in [19]

Before illustrating the cryptosystem in [19], we first describe the necessary Euler's Theorem. For any integers a, n, there is

$$a^{\varphi(n)} \equiv 1 \bmod n \text{if } \gcd(a, n) = 1,$$

where $\varphi(n)$ is the Euler function of the integer n.

The cryptosystem used in [19] is from two cyclic multiplicative group $G_1 = <g_1>, G_2 = <g_2>$ of order q. The two generators are mainly generated from the above Euler's Theorem: randomly choose two large primes p, q such that q divides $p-1$ (i.e., $q|p-1$), then the generators are described as follows:

$$g_1 = h^{\frac{p-1}{q}} \bmod p \text{ and } g_2 = g_1^p \bmod p^2,$$

where $h \in \mathbb{Z}_p \backslash \{0\}$. The public parameters are denoted as

$$pp = (G_1, G_2, g_1, g_2, p, p^2, q).$$

Definition 1 (*Computational discrete algorithm* (*CDH*) *problem*) [23] Given only $g, g^a, g^b \in G$, where g is the generator of a multiplicative group G and $a, b \in \mathbb{Z}$, computing g^{ab} without knowing a, b is computationally intractable.

2.3 Distributed paillier cryptosystem

The Paillier cryptosystem [24] is a public key cryptographic scheme with additive homomorphism proposed by Paillier in 1999. The distributed Paillier cryptosystem is an improved version described as follows, which includes key generation (KeyGen), encryption (Enc), decryption (Dec) and partial decryption (Partial Dec) algorithm.

- KeyGen: taking the security parameter as input, randomly choose two large primes p, q, compute $N = pq$ and $\lambda = \text{lcm}(p-1, q-1)$. Randomly choose an integer $g \in \mathbb{Z}_{N^2}^*$, then the order of g is N. Note that $\gcd(\lambda, N^2) = 1$, then there exists $\lambda_1, \lambda_2, \ldots, \lambda_n$ for random n such that

$$\begin{cases} \lambda_1 + \lambda_2 + \cdots + \lambda_n \equiv 0 \bmod \lambda \\ \lambda_1 + \lambda_2 + \cdots + \lambda_n \equiv 1 \bmod N^2. \end{cases}$$

  The public key is $pk = (N, g)$ and the secret key is $sk = (\lambda, \lambda_1, \lambda_2, \ldots, \lambda_n)$.
- Enc: randomly choose $r \in \mathbb{Z}_{N^2}^*$ to encrypt the plaintext $\mu \in \mathbb{Z}_N$ to produce a ciphertext

$$E_{pk}(\mu) = c = g^{\mu} r^N \bmod N^2.$$

- Dec: Upon receipt of a ciphertext c, compute as Paillier cryptosystem

$$D_{sk}(c) = \mu = \frac{L(c^{\lambda} \bmod N^2)}{L(g^{\lambda} \bmod N^2)} \bmod N,$$

  where $L(x) = \frac{x-1}{N}$ for integer x.
- Partial Dec: Note that $\lambda_1 + \lambda_2 + \cdots + \lambda_n \equiv 0 \bmod \lambda$, then the i-th partition performs c^{λ_i} for $i \in [n]$. After that, compute

$$c' = \prod_{i=1}^{n} c^{\lambda_i}.$$

  Recall that $\lambda_1 + \lambda_2 + \cdots + \lambda_n \equiv 1 \bmod N^2$, then the plaintext can also be recovered by

$$D_{sk}(c) = \mu = \frac{L(c' \bmod N^2)}{L(g \bmod N^2)} \bmod N.$$

Correctness Now we will illustrate the correctness of partial decryption algorithm. Since $\lambda_1 + \lambda_2 + \cdots + \lambda_n \equiv 0 \bmod \lambda$, there is

$$c' \bmod N^2 = c^{\lambda} \bmod N^2.$$

Moreover, since $g \in \mathbb{Z}_{N^2}^*$ and the order of g is N, there exists $t \in (0, N)$ such that $g = 1 + tN$. Note that

$$\begin{cases} \lambda_1 + \lambda_2 + \cdots + \lambda_n \equiv 0 \bmod \lambda \\ \lambda_1 + \lambda_2 + \cdots + \lambda_n \equiv 1 \bmod N^2, \end{cases}$$

we have

$$\begin{aligned} g^{\lambda} \bmod N^2 &= (1+tN)^{N^2+1} \bmod N^2 \\ &= 1 + tN \bmod N^2 \\ &= g \bmod N^2. \end{aligned}$$

Therefore, the partial decryption algorithm holds dependent on the decryption of Paillier cryptosystem.

Homomorphism and security The distributed Paillier cryptosystem has the following properties:

1) Homomorphic addition

- Given $E_{pk}(\mu_a)$ and $E_{pk}(\mu_b)$ with underlying plaintexts μ_a, μ_b respectively, we can compute
$D_{sk}\big(E_{pk}(\mu_a) \cdot E_{pk}(\mu_b)\big) = \mu_a + \mu_b.$
- Given a plaintext $\alpha \in \mathbb{Z}_N$, we can compute
$D_{sk}\big(\big(E_{pk}(\mu_a)\big)^{\alpha}\big) = \alpha \cdot \mu_a.$

2) Semantic security

The security of distributed Paillier cryptosystem depends on the Paillier cryptosystem under the Composite Residuosity Class Problem [24]. Then the distributed Paillier cryptosystem also has the semantic security, that is, given a set of ciphertexts, the probabilistic polynomial time adversary cannot infer any information about the corresponding plaintexts.

3 Cryptanalysis of the PPDARM in [19]

In this section, we describe the PPDARM in [19], followed by some cryptanalysis.

3.1 The PPDARM in [19]

The PPDARM in [19] serves to compute $C_{count} = \sum_{i=1}^{n} C_{count(i)}$ from n honest-but-curious EHR systems without disclosing the private value $C_{count(i)} \in G_1$ of the i-th EHR system for $i \in [n]$. Specifically, every EHR system starts with finding 1-frequent itemsets from the original dataset with Apriori algorithm locally. Then PPDARM consisting of the set-up phase, encryption phase and sum phase is presented.

- Set-up Phase: Taking as input the random integer $r_i \in \mathbb{Z}_q$ for $i \in [n]$, every EHR system shares with the adjacent two EHR systems, and obtains the random integer $R_i \in G_2$. The details are described in Algorithm 1. Note that, even if R_i is obtained in this phase, the adversary cannot recover r_i due to the CDH assumption.
- Encryption Phase: The i-th EHR system for $i \in [n]$ computes the ciphertext using his own random integer R_i to encrypt $C_{count(i)}$ and broadcasts the ciphertext to all EHR systems as Algorithm 2.
- Sum Phase: Refer to Algorithm 3. Upon receiving the ciphertext Y_i, every EHR system computes the modular multiplication of all the ciphertexts as well as a function operation.

The correctness of PPDARM is omitted here and refer to [19] for more details. But we must emphasize that only $C_{count} = \sum_i C_{count(i)} < p$ holds for

$$1 + pC_{count} \bmod p^2 = 1 + pC_{count},$$

the correctness follows as desired.

Algorithm 1: Set-up Algorithm (Every EHR System)

Input: Public parameters $pp = (G_2, g_2, p^2, q)$ described as Section 2.2.

1 Randomly chooses $r_i \in \mathbb{Z}_q$, the i-th EHR system computes $X_i = g_2^{r_i} \in G_2$ for $i \in [n]$.

2 The i-th EHR system shares X_i with the $(i-1)\bmod n$-th and $(i+1)\bmod n$-th EHR systems as shown in Fig 1(a), and computes $R_i = (g_2^{r_{i+1}} / g_2^{r_{i-1}})^{r_i} \bmod p^2$.

Output: R_i.

Algorithm 2: Encryption Algorithm (Every EHR System)

Input: The plaintext $C_{count(i)}$.

1 Computes $Y_i = \big(1 + pC_{count(i)}\big)\, R(i) \bmod p^2$ and broadcasts it to all EHR systems as Fig 1(b).

Output: Y_i.

Algorithm 3: Sum Algorithm (Every EHR System)

Input: The ciphertext Y_i.

1 Every EHR system computes $Y = \prod_{i=1}^{n} Y_i \bmod p^2$, outputs $C_{count} = \frac{Y-1}{p} \bmod p$.

Output: C_{count}.

3.2 Our cryptanalysis of the PPDARM in [19]

In this subsection, we first present some mistakes about typos and theoretical performance analyses in [19], then an effective attack algorithm against the PPDARM is elaborated.

(1) The length of parameters p, q.

In terms of the parameters $pp = (G_1, G_2, g_1, g_2, p, p^2, q)$ in [19], they claimed $|p|_2 = |q|_2$ and $q|p-1$. However, the two conditions are contradiction. If $q|p-1$, there is $q \leq p-1 < p$, then $|q|_2 \leq |p|_2$ not $|p|_2 = |q|_2$ always holds.

(2) The theoretical performance analyses of the PPDARM.

The revised theoretical performance analysis in [19] is illustrated as Table 2 and Table 3. We review some basic conclusions before analyzing the computational complexity of the PPDARM in [19].

1) The computational complexity of modular exponentiation modulo p is $O(\log^3 p)$.

2) The computational complexity of modular inverse modulo p is $O(\log^3 p)$.

3) The computational complexity of the multiplication modulo p is $O(\log^2 p)$.

- Through the above descriptions, we can see that the computational costs for every EHR system in Algorithm 1 are from a division and a modular exponentiations modulo p^2 (or a modular inverse and two modular exponentiations modulo p^2 since the algebra structure is finite field of order p^2), that is, $O(\log^3 p)$, rather than $O(q)$ from $r_i \in \mathbb{Z}_q$ as Domadiya et al. claimed. Similarly, the communication complexity for every EHR system in this phase depends on two interactions, which can be seen as $O(1)$. And the corresponding communication overhead for every EHR system is 2 times length of X_i, i.e., $2|X_i|_2 = 4\log p$.
- In Algorithm 2, every EHR system performs two modular multiplications modulo p^2 and negligible modular addition, then the computational complexity is $O(\log^2 p)$, which can also be regarded as $O(1)$ since p is relevant to the security parameter. With respect to the Encryption Phase, it requires the broadcast of the ciphertext to $n-1$ EHR systems. Then the communication complexity for every EHR system in this phase is $O(n)$, and the communication overhead for every EHR system is $n-1$ times length of Y_i, i.e., $(n-1)|Y_i|_2 = 2(n-1)\log p$.
- After receiving the ciphertexts of all other EHR systems, every EHR system performs a division of function $L(\cdot)$ and $n-1$ modular multiplications modulo p^2, where the number of modular multiplications can be reduced by binary tree in parallel. Note that the practicability in real life for $n << p$, the computational complexity in Algorithm 3 can be represented as $O(\log^2 p)$. But in the opinion of Domadiya et al., they only focused on the number of EHR systems even if $n << p$, which is unreasonable. In terms of communication complexity, every EHR system just performs $n-1$ modular multiplications modulo p^2 in local, as Domadiya et al. analyzed this phase is performed without interactions. Hence the communication complexity is none instead of $O(1)$, and likewise for communication overhead.

Table 2 Theoretic comparisons of PPDARM for every EHR system in [19]

Phase	Our revised analysis			Initial analysis [19]	
	Computational complexity	Communication complexity	Communication overhead	Computational complexity	Communication complexity
Set-up	DI+ME	$O(1)$	$2\|X_i\|_2$	$O(q)$ (×)	$O(1)$
Encryption	2MM+MA	$O(n)$	$(n-1)\|Y_i\|_2$	$O(1)$	$O(n)$
Sum	$\lceil \log (n-1) \rceil$MM+DI	—	—	$O(n)$ (×)	$O(1)$ (×)

MI: modular inverse modulo p^2. ME: modular exponentiation modulo p^2. DI: division.

MM: modular multiplication modulo p^2. MA: modular addition modulo p^2

(3) The attack algorithm against the PPDARM

The PPDARM in [19] aims to compute the sum of the supports C_{count} from n semi-honest EHR systems without compromising the privacy of the i-th EHR system $C_{count(i)}$ for $i \in [n]$. Although Domadiya et al. claimed the scheme is secure under CDH assumption in G_2, we can still recover any $C_{count(i)}$ only by a modular exponentiation as shown in Algorithm 4.

Algorithm 4: Attack Algorithm

Input: The ciphertext $Y_i = \left(1 + pC_{count(i)}\right) R(i) \bmod p^2$ and the public parameter $pp = (p, p^2, q)$ of G_2.

1 Upon the receipt of the ciphertext Y_i, compute $C_{count(i)} = \frac{Y_i^q \bmod p^2 - 1}{pq} \bmod p$.

Output: $C_{count(i)}$.

Correctness. It is clear that

$$\begin{aligned} Y_i^q \bmod p^2 &= \left(1 + pC_{count(i)}\right)^q \left(g_2^{r_{i+1}} / g_2^{r_{i-1}}\right)^{qr_i} \bmod p^2 \\ &= \left(1 + pC_{count(i)}\right)^q g_2^{qr_i(r_{i+1} - r_{i-1})} \bmod p^2 \\ &= 1 + pqC_{count(i)} \bmod p^2 \cdot \because g_2^q = 1 \bmod p^2 \end{aligned}$$

When $C_{count(i)} < \frac{p^2-1}{pq}$ and $C_{count(i)} \in \mathbb{Z}_p$ holds, we have

$$1 + pqC_{count(i)} \bmod p^2 = 1 + pqC_{count(i)}.$$

Then the support $C_{count(i)}$ for the i-th EHR system can be recovered by

$$C_{count(i)} = \frac{Y_i^q \bmod p^2 - 1}{pq} \bmod p.$$

4 Our PPDARM scheme

In order to protect the support sum of all the EHR systems without leaking their privacy, we propose a new PPDARM scheme with less interactions among EHR systems compared with [19]. The problem description, model, scheme, security analysis and performance evaluation of our proposal are presented.

4.1 Problem description

The EHR system can be thought as carte which can not only store a large number of medical data but also perform some computations upon the data. There are n number of EHR systems involved in our PPDARM scheme. The message $C_{count(i)} \in \mathbb{Z}_N (i \in [n])$ for the i-th EHR system is produced in the preprocessing phase, where N is a public parameter as shown in Sect. 2.3. The Health Committees wants to obtain the support sum $C_{count} = \sum_i C_{count(i)}$ through a mining scheme, to know about the heart disease to make measures for healthcare service management. The mining scheme is required not to leak the information of mining results and all $C_{count(i)}$s of EHR systems.

4.2 Our model

This section is composed of system model, threat model and design goals of our PPDARM scheme.

4.2.1 System model

Our system model consists of three entities: Key Generation Center, EHR Systems and Miner, as shown in Fig. 2.

- Key Generation Center (KGC): a trusted institution. KGC randomly generates and distributes the keys for every EHR system and Miner during the set-up phase.
- EHR Systems: carte containing some medical symptom sets. EHR Systems search and calculate the local support of the medical symptom sets, and take them as private information. They also perform auxiliary data mining tasks for Miner without giving in the privacy of their own supports.
- Miner: the Health Committee. Miner can query the disease by uploading physical symptoms and obtain some mining results to effectively predict the disease.

In the preprocessing phase, the Miner makes a query to the EHR Systems, all the EHR Systems locally compute the support of their own data. At the same time, the KGC randomly distributes keys to the EHR Systems and Miner. Then EHR Systems can encrypt the supports with the public key from KGC and send corresponding ciphertexts back to Miner. After some computations of Miner, all EHR Systems provide some partial decryptions for Miner. At the end of the execution, Miner obtains a mining result without knowing the relevant information of all the EHR Systems.

Table 3 Theoretic comparisons of PPDARM

Proposed approach						Existing approach [19]			
Phase	Computational complexity		Communication complexity		Communication overhead	Phase	Computational complexity	Communication complexity	Communication overhead
	EHR system	Miner	EHR system	Miner	EHR system	Miner		EHR system	EHR system
Set-up	–	–	–	–	–	–	Set-up	$O(\log^3 p)$	$O(1)$	$4\log p$
Encryption	$O(\log^3 N)$	$O(\log^2 N)$	$O(1)$	–	$2\log N$	–	Encryption	$O(\log^2 p)$	$O(n)$	$2(n-1)\log p$
Decryption	$O(\log^3 N)$	$O(\log^3 N)$	$O(1)$	$O(n)$	$2\log N$	$2n\log N$	Sum	$O(\log^2 p)$	–	–

4.2.2 Threat model

In our scheme, all entities fully trust KGC. The EHR Systems and Miner are honest but curious, which means that they strictly follow the scheme and return the correct results, but still are curious and try to infer some private information upon the received information. Moreover, there is no collusion among EHR Systems and Miner, let alone EHR Systems. Any probabilistic polynomial time attacker $\mathcal{A}$ attempts to obtain the supports of the EHR Systems and the mining results of Miner from all data transmitted on an insecure channel.

Someone may doubt our system model owning to the trusted institution KGC. The communication channel where the KGC randomly distributes keys to the EHR Systems and Miner must be secure to avoid adversaries from eavesdropping the secret keys of EHR Systems and Miner, which seems too ideal. In fact, even if there is no trusted institution KGC, the certificate is required to verify the identity of the EHR Systems in [19]. However, Domadiya et al. paid no attention to this problem and only asked the EHR Systems to be semi-honest. Therefore, the pseudo EHR Systems can avail themselves of the opportunity to get in the mining scheme playing the role of semi-honest EHR Systems, and no one can penetrate their identities, which causes a big security threat. Contrarily, in our system model, the process of distributing keys to the EHR Systems and Miner for KGC can be seen as a "black box", we only care about the outputs, that is, EHR Systems and Miner possess the public key and their own secret keys private for any others. The problem that every EHR System shares the same parameters or secret keys will be resolved.

4.2.3 Design goals

Under the above system model and threat model, our design goals are as follows.

- Privacy. The privacy of EHR Systems should be protected effectively during the mining process. That is, the support of every EHR System should be private for any others. In addition, only Miner can obtain the final mining results.
- Accuracy. The privacy of the mining results should be ensured, when the mining results are also required to be accurate.
- Feasibility. The performance of the privacy-preserving distributed association rule mining scheme based on medical data should be efficient, which can be applied to real life.

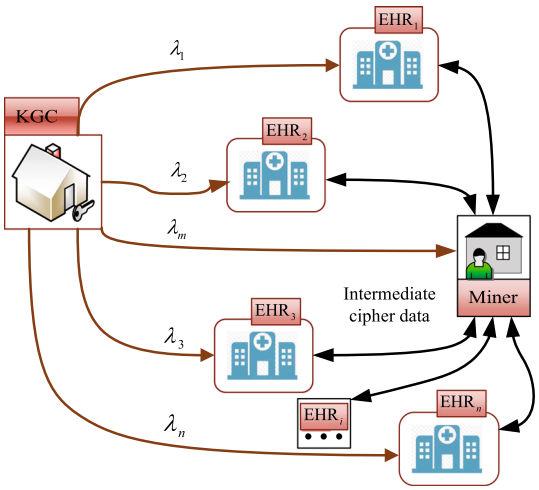


Fig. 2 Our system model

4.3 Our mining scheme

Before illustrating our mining scheme consisting of set-up phase, encryption phase and decryption phase, we first present a preprocessing phase to produce the support $C_{count(i)}$ for $i \in [n]$ of the i-th EHR System. That is, Miner makes a query to every EHR System, then the i-th EHR System computes locally his own data and produces a plaintext denoted by $C_{count(i)}$ for all $i \in [n]$. Now our mining scheme is unfolded as follows.

- Set-up Phase: The KGC randomly distributes public key and secret keys to the EHR Systems and Miner, as shown in Algorithm 5.
- Encryption Phase: Every EHR system sends a ciphertext using the encryption algorithm of distributed Paillier cryptosystem as Sect. 2.3 to Miner. The details are described in Algorithm 6.
- Decryption Phase: Refer to Algorithm 7. Upon receiving the ciphertext Y, every EHR system decrypts it with his own secret key and sends the decrypted result back to Miner. After some computations, Miner obtains the final mining result.

Correctness It is clear that for all $i \in [n]$

$$Y = g^{\sum_i C_{count(i)}} \left(\prod_i r_i\right)^N \bmod N^2,$$

and

$$\begin{aligned} DC_{count} &= \left(Y^{\lambda_m} \cdot \prod_{i=1}^{n} DY_i\right) \bmod N^2 \\ &= g^{\left(\lambda_m + \sum_i \lambda_i\right)\sum_i C_{count(i)}} \left(\prod_i r_i\right)^{N\left(\lambda_m + \sum_i \lambda_i\right)} \bmod N^2 \\ &\quad \because \lambda_m + \sum_i \lambda_i \equiv 0 \bmod \lambda \text{ and } r_i^{N\lambda} \equiv 1 \bmod N^2 \\ &= g^{\left(\lambda_m + \sum_i \lambda_i\right)\sum_i C_{count(i)}} \bmod N^2. \end{aligned}$$

Therefore, we have

$$\begin{aligned} &\frac{L(DC_{count} \bmod N^2)}{L(g \bmod N^2)} \bmod N \\ &= \frac{L\left((1+tN)^{\left(\lambda_m + \sum_i \lambda_i\right)\sum_i C_{count(i)}} \bmod N^2\right)}{L(1+tN \bmod N^2)} \bmod N \\ &= \frac{tN\left(\lambda_m + \sum_i \lambda_i\right)\sum_i C_{count(i)} \bmod N^2}{tN \bmod N^2} \bmod N \\ &\quad \because \lambda_m + \sum_i \lambda_i \equiv 1 \bmod N^2 \\ &= \frac{tN \sum_i C_{count(i)} \bmod N^2}{tN \bmod N^2} \bmod N \\ &= \sum_i C_{count(i)} = C_{count}. \end{aligned}$$

4.4 Security analysis

We analyze the security of our scheme with insecure channel under the threat model introduced in Sect. 4.2.2. Assume that the adversary $\mathcal{A}$ interacts with the challenger who has secret information in the real world and the simulator $\mathcal{S}$ in the ideal world. In our scheme, we regard the

Algorithm 5: Set-up Algorithm (KGC+Every EHR System +Miner)

Input: The security parameter.

1 Taking the security parameter as input, KGC performs the key generation algorithm of distributed Paillier cryptosystem as Section 2.3, and distributes secret keys $\lambda_1, \lambda_2, \ldots, \lambda_n, \lambda_m$ such that

$$\begin{cases} \lambda_1 + \lambda_2 + \cdots + \lambda_n + \lambda_m \equiv 0 \bmod \lambda \\ \lambda_1 + \lambda_2 + \cdots + \lambda_n + \lambda_m \equiv 1 \bmod N^2 \end{cases}$$

to the n EHR Systems and Miner respectively.

Output: The public key is $pk = (N, g)$.

Algorithm 6: Encryption Algorithm (Every EHR System+Miner)

Input: The public key is $pk = (N, g)$.

1 Randomly chooses $r_i \in \mathbb{Z}^*_{N^2}$, the i-th EHR System computes $Y_i = g^{C_{count(i)}} r_i^N \bmod N^2$ for all $i \in [n]$.

2 Every EHR System sends Y_i to Miner, and Miner performs modular multiplication as $Y = \prod_{i=1}^{n} Y_i \bmod N^2$.

Output: Y.

Algorithm 7: Decryption Algorithm (Every EHR System+Miner)

Input: The ciphertext Y.

1 After receiving the ciphertext Y from Miner, the i-th EHR system computes $DY_i = Y^{\lambda_i} \bmod N^2$ for $i \in [n]$, and sends DY_i back to Miner.

2 Miner performs $DC_{count} = \left(Y^{\lambda_m} \cdot \prod_{i=1}^{n} DY_i\right) \bmod N^2$, and obtains the final mining result by

$$C_{count} = \frac{L\left(DC_{count} \bmod N^2\right)}{L\left(g \bmod N^2\right)} \bmod N.$$

Output: C_{count}.

EHR System or Miner as challenger in different phases. If the view in the real world is computationally indistinguishable from the one in the ideal world, that is,

$$\left\{\mathrm{REAL}_{\mathcal{A}}^{\text{EHR System}}(\cdot)\right\} \stackrel{c}{\equiv} \left\{\mathrm{IDEAL}_{\mathcal{S}}^{\text{EHR System}}(\cdot)\right\}$$

$$\left\{\mathrm{REAL}_{\mathcal{A}}^{\text{Miner}}(\cdot)\right\} \stackrel{c}{\equiv} \left\{\mathrm{IDEAL}_{\mathcal{S}}^{\text{Miner}}(\cdot)\right\}$$

then we say our mining scheme secure.

Theorem 1 *Our data mining scheme is secure under semi-honest model.*

Proof The security in the Set-up phase depends on the trusted institution KGC. We focus on the security analysis of the Encryption and Decryption phase. □

Lemma 1 *During the Encryption phase, every EHR System is secure against a semi-honest adversary* $\mathcal{A}_{\mathcal{S}}^{\text{EHR System}}$ *corrupting the EHR System in the real world, likewise for Miner against a semi-honest adversary* $\mathcal{A}_{\mathcal{S}}^{\text{Miner}}$.

Proof The view of the i-th EHR System for $i \in [n]$ in this phase contains Y_i, i.e., the encryption of secret support $C_{count(i)}$. And the view of Miner are the n ciphertexts of supports uploaded from every EHR System and the multiplication of these ciphertexts Y, i.e., $Y_1, \ldots, Y_n, Y$.

Considering that our threat model is semi-honest, every EHR System chooses a random integer to encrypt his own support. Moreover, the credible KGC distributes secret keys to every EHR System and Miner randomly, any entity knows nothing about any other secret keys. Coupled with no collusion of our threat model, the security of distributed Paillier cryptosystem ensures that the i-th support is private for j-th EHR System, let alone Miner, where $i \in [n]$ and $j \in [n] \backslash \{i\}$. In other words, for $i \in [n]$,

$$\left\{\mathrm{REAL}_{\mathcal{A}}^{\text{EHR System}}(Y_i)\right\} \stackrel{c}{\equiv} \left\{\mathrm{IDEAL}_{\mathcal{S}}^{\text{EHR System}}(Y_i)\right\}.$$

$$\left\{\mathrm{REAL}_{\mathcal{A}}^{\text{Miner}}(Y_i)\right\} \stackrel{c}{\equiv} \left\{\mathrm{IDEAL}_{\mathcal{S}}^{\text{Miner}}(Y_i)\right\}.$$

The Miner also performs the modular multiplication after receiving the n ciphertexts uploaded from EHR Systems. Due to semi-honest Miner, we can easily conclude that the Miner is secure against a semi-honest adversary $\mathcal{A}_{\mathcal{S}}^{\text{Miner}}$ corrupting the Miner in the real world, that is,

$$\left\{\mathrm{REAL}_{\mathcal{A}}^{\text{Miner}}(Y_1, \ldots, Y_n, Y)\right\} \stackrel{c}{\equiv} \left\{\mathrm{IDEAL}_{\mathcal{S}}^{\text{Miner}}(Y_1, \ldots, Y_n, Y)\right\}.$$

□

Lemma 2 *During the Decryption phase, every EHR System and Miner are secure against honest-but-curious* $\mathcal{A}_{\mathcal{S}}^{\text{EHR System}}$ *and* $\mathcal{A}_{\mathcal{S}}^{\text{Miner}}$ *in the real world respectively.*

Proof The semi-honest model requires every EHR System and Miner to strictly follow algorithm 7 and output correct results. The restriction of no collusion among EHR Systems and Miner makes it impossible for probabilistic polynomial time adversary to obtain the essential sum of

secret keys $\lambda_1 + \lambda_2 + \cdots + \lambda_n + \lambda_m$. That is, every EHR System cannot know the final mining result as Miner, i.e.,

$$\left\{\text{REAL}_{\mathcal{A}}^{\text{EHR System}}(Y)\right\} \stackrel{c}{\equiv} \left\{\text{IDEAL}_{\mathcal{S}}^{\text{EHR System}}(Y)\right\}$$

and

$$\left\{\text{REAL}_{\mathcal{A}}^{\text{EHR System}}(DY_i)\right\} \stackrel{c}{\equiv} \left\{\text{IDEAL}_{\mathcal{S}}^{\text{EHR System}}(DY_i)\right\}$$

due to the distributed Paillier cryptosystem.

In terms of Miner, as he honestly performs the Algorithm 7, with the help of other semi-honest EHR systems, the final mining result can be acquired by him as desired.

According to the two lemmas described above, we show that our scheme is secure. For the Miner, he learns nothing except the support sum of the EHR Systems. In terms of every EHR System, they learns nothing about the supports of any other EHR Systems and the mining result of Miner. □

4.5 Performance evaluation

In this section, we illustrate the efficiency of our PPDARM scheme from the perspective of theoretical analysis and experimental analysis.

4.5.1 Theoretical analysis

The theoretical performance of computational complexity, communication complexity and communication overhead is shown in Table 3.

- As the Set-up Phase describes, only the trusted institution KGC randomly distributes public key and secret keys to every EHR system and Miner, then the computational complexity, communication complexity and communication overhead for every EHR system and Miner are all none.
- During the Encryption Phase, every EHR System encrypts his own support with distributed Paillier cryptosystem and sends the ciphertext to Miner, and the Miner performs $n-1$ number of modular multiplications modulo N^2, which can be reduced $\lceil \log (n-1) \rceil$ number of modular multiplications with dichotomy, i.e., with computational complexity $O(\log^2 N)$. Considering the encryption algorithm of distributed Paillier cryptosystem as described in Sect. 2.3, the computational complexity for every EHR system is two modular exponentiations modulo N^2 and a modular multiplication modulo N^2, that is, $O(\log^3 N)$. In terms of the communication complexity, every EHR System sends his ciphertext to Miner, so the communication complexity and communication overhead for every EHR System are $O(1)$ and $|Y_i|_2 = 2\log N$ respectively. Miner has no interaction with EHR Systems and only performs modular multiplications, so the communication complexity and communication overhead for every EHR System are both empty.
- In Algorithm 7, Miner sends Y to every EHR system to seek for partial decryption. In turn, every EHR system transmits his result to Miner; Miner also decrypts Y with his own secret key and multiplies the $n+1$ results. A division operation ends the algorithm and Miner obtains the final mining result. For every EHR system, the computational complexity is a modular exponentiation with $O(\log^3 N)$; he makes interactions with Miner with communication complexity $O(1)$ and communication overhead $|DY_i|_2 = 2\log N$ respectively. As for Miner, he performs modular exponentiation, n modular multiplications and division, of computational complexity $O(\log^3 N)$. He makes interactions with n EHR systems, so the communication complexity and communication overhead for Miner are $O(n)$ and $n|Y|_2 = 2n\log N$ respectively.

From Table 3, we can easily see that the theoretical performance of ours and [19] are different due to the system model, that is, we assign the Miner to be the unique insider who knows the final mining results while [19] makes every EHR system obtain the final mining results. From the perspective of the whole PPDARM scheme, the computational, communication complexity and communication overhead for every EHR system in ours is relatively smaller than in [19], due to $|N|_2 = |p|_2 \geq 2048$ bits for the consideration of security in real life. Coupled with the security analysis in Sects. 3.2 and 4.4, our PPDARM scheme is more efficient and practical.

4.5.2 Experimental analysis

In this section, we show the performance of our scheme by conducting tests on personal computers utilizing the NTL library [25]. The environment is listed as follows:

- CPU: Intel(R) Core(TM) i5-8300H 2.30GHz
- RAM: 8.00GB
- OS: Windows 10

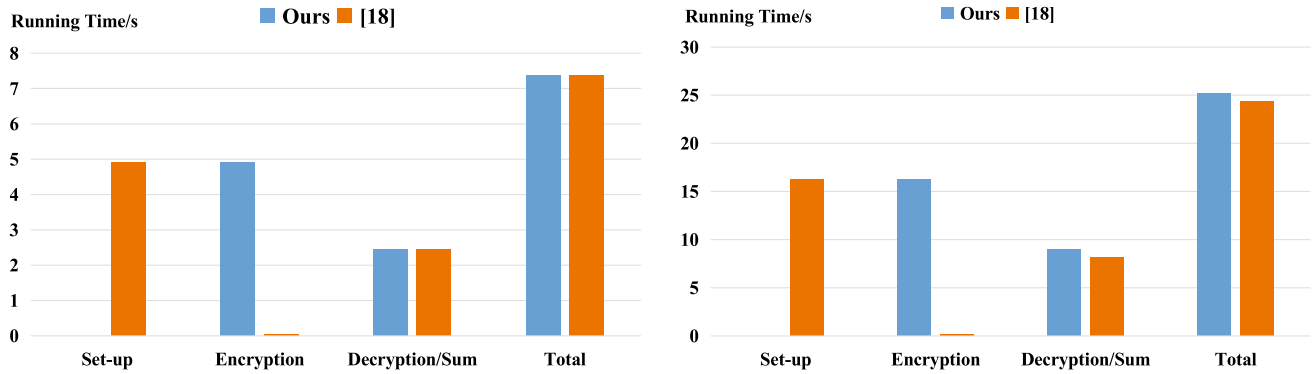
The datasets are from [26] following the routine of [19]. We focus on the 14 attributes affecting the heart disease as Domadiya et al. claimed. In our experiment, the datasets in [26] are all composed of 0, 1. For example, the third attribute "Type of chest pain: (1=typical angina, 2=atypical angina, 3=non-anginal pain, 4=asymptomatic)", we make a transfer: "1=typical angina" $\Leftrightarrow (1,0,0,0)$, "2=atypical angina" $\Leftrightarrow (0,1,0,0)$,"=non-anginal pain" $\Leftrightarrow (0,0,1,0)$,

Table 4 Average running time of common operations

$\|N\|_2$	Modular multiplication modulo N	Modular exponentiation modulo N
2048 bits	1.125ms	2458.032ms
3072 bits	2.452ms	8099.641ms

Table 5 Comparisons of computation cost with $|N|_2 = |p|_2 = 2048, 3072$ Bits for $n = 5$

Proposed approach					Existing approach [19]		
Phase	$\|N\|_2 = 2048$ bits		$\|N\|_2 = 3072$ bits		Phase	$\|p\|_2 = 2048$ bits	$\|p\|_2 = 3072$ bits
	EHR Systems	Miner	EHR Systems	Miner		EHR Systems	EHR Systems
Set-up	0s	0s	0s	0s	Set-up	4.916s	16.199s
Encryption	4.917s	2.250ms	16.202s	4.904ms	Encryption	2.255ms	4.905ms
Decryption	2.458s	4.919s	9.000s	16.207s	Sum	2.460s	8.105s
Total	7.375s	4.921s	25.202s	21.133s	Total	7.378s	24.309s



(a) Comparisons with $|N|_2 = |p|_2 = 2048$ Bits for $n = 5$.

(b) Comparisons with $|N|_2 = |p|_2 = 3072$ Bits for $n = 5$.

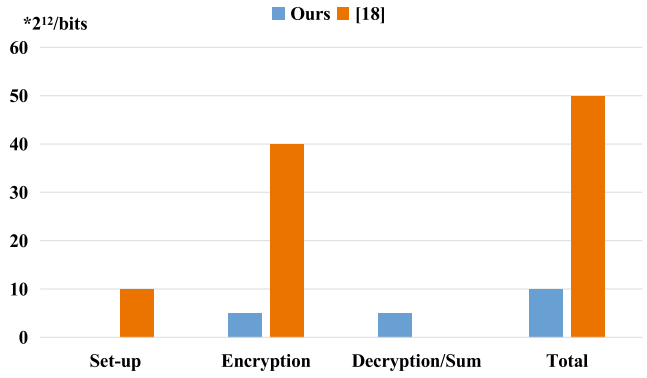
Fig. 3 Comparisons of computation cost

Table 6 Comparisons of communication overhead

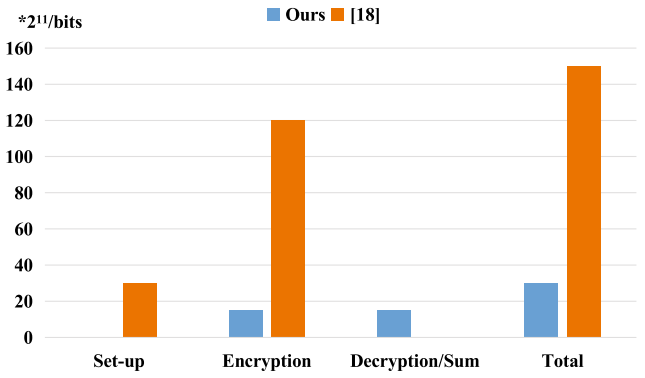
Proposed approach					Existing approach [19]		
	$\|N\|_2 = 2048$ bits		$\|N\|_2 = 3072$ bits			$\|p\|_2 = 2048$ bits	$\|p\|_2 = 3072$ bits
Phase	EHR systems	Miner	EHR systems	Miner	Phase	EHR systems	EHR systems
Set-up	–	–	–	–	Set-up	$2^{13}n$ bits	$3 \cdot 2^{12}n$ bits
Encryption	$2^{12}n$ bits	–	$3 \cdot 2^{11}n$ bits	–	Encryption	$2^{13}n(n-1)$ bits	$3 \cdot 2^{12}n(n-1)$ bits
Decryption	$2^{12}n$ bits	$2^{12}n$ bits	$3 \cdot 2^{11}n$ bits	$3 \cdot 2^{11}n$ bits	Sum	–	–
Total	$2^{13}n$ bits	$2^{12}n$ bits	$3 \cdot 2^{12}n$ bits	$3 \cdot 2^{11}n$ bits	Total	$2^{13}n^2$ bits	$3 \cdot 2^{12}n^2$ bits

and "4=asymptomatic" $\Leftrightarrow (0, 0, 0, 1)$, likewise for other attributes. The dimensions of the datasets enlarged a little without changing the values of the heart disease. Under this circumstance, the performance comparisons are evaluated.

To achieve the security requirement, we choose $|N|_2 = |p|_2 \geq 2048$ bits in distributed Paillier cryptosystem and [19] respectively to ensure security. Some common operations are first evaluated in Table 4. We set the number of EHR Systems ranging from 4 to 10 as Domadiya et al. [19]



(a) Comparisons with $|N|_2 = |p|_2 = 2048$ Bits for $n = 5$.



(b) Comparisons with $|N|_2 = |p|_2 = 3072$ Bits for $n = 5$.

Fig. 4 Comparisons of communication overhead

to test the performance of our proposal and [19] including computation cost and communication overhead.

Computation cost. The comparison of running time between our proposal and [19] is shown in Table 5 with $|N|_2 = |p|_2 = 2048, 3072$ bits for random $n = 5$. Considering that every EHR System can perform operations in parallel, the running time for every EHR System is also corresponding to the one for all EHR Systems. We also represent the comparisons of computation cost through bar graph as shown in Fig. 3. We claim that Fig. 3 clearly shows the running time of our proposed scheme and [19] are almost equal for EHR Systems. In terms of the time for Miner, it is a little more for ours than [19]. From these results, our PPDARM scheme is approximately efficient as the existing scheme in the literature.

Communication overhead. The communication overhead of our proposal and [19] is presented in Table 6, which clearly shows that our scheme yields less interactions when comparing to [19]. We represent the comparison of communication overhead through a bar graph as shown in Fig. 4. It is apparent that our PPDARM scheme is more efficient in terms of communication overhead.

5 Conclusion

In this paper, we focus on the privacy-preserving distributed association rule mining scheme based on medical data. We first present the weakness of Domadiya et al.'s PPDARM scheme: 1) the support of the itemsets from every EHR can be easily recovered by a modular exponentiation; 2) the performance analyses are not exactly correct. On the basis of this situation, the distributed Paillier cryptosystem is utilized to construct our PPDARM scheme, avoiding the same flaws of Domadiya et al.. Performance evaluations including theoretical analysis and experimental analysis demonstrate our proposal is more efficient and feasible, especially in communication.

Acknowledgements This research was funded by the National Key R&D Program of China under Grant No. 2017YFB0802000, the National Natural Science Foundation of China under Grant Nos. U19B2021, 61972457, the National Cryptography Development Fund under Grant No. MMJJ20180111, and Key Research and Development Program of Shaanxi under Grant No. 2020ZDLGY08-04.

Declarations

Conflict of interest Authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants performed by any of the authors.

References

1. Azees, M., Vijayakumar, P., Karuppiah, M., & Nayyar, A. (2021). An efficient anonymous authentication and confidentiality preservation schemes for secure communications in wireless body area networks. *Wireless Networks, 27*(3), 2119–2130.
2. Bhatia, S., C, P., & Dey, N. (2020). Data mining and information retrieval. Opinion Mining. *Information Retrieval*
3. Zhang, L., Wang, W., & Zhang, Y. (2019). Privacy preserving association rule mining: Taxonomy, techniques, and metrics. *IEEE Access, 7,* 45032–45047.
4. Thabtah, F. A. (2007). A review of associative classification mining. *Knowledge Engineering Review, 22*(1), 37–65.
5. Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: A review. *ACM Computing Surveys, 31*(3), 264–323.
6. Deogun, J. S., & Jiang, L. (2005). Prediction mining—an approach to mining association rules for prediction. In: D. Slezak, J. Yao, J. F. Peters, W. Ziarko, X. Hu (Eds.), *Rough sets, fuzzy sets, data mining, and granular computing, 10th international conference, RSFDGrC 2005, Regina, Canada, August 31–*

September 3, 2005, Proceedings, Part II, *Lecture Notes in Computer Science* (Vol. 3642, pp. 98–108). Springer.
7. Ma, C., Wang, B., Jooste, K., Zhang, Z., & Ping, Y. (2020). Practical privacy-preserving frequent itemset mining on supermarket transactions. *IEEE Systems Journal, 14*(2), 1992–2002.
8. Ordonez, C. (2006). Association rule discovery with the train and test approach for heart disease prediction. *IEEE Transactions on Information Technology in Biomedicine, 10*(2), 334–343.
9. Shin A. M., Lee I. H., & G. H. L. E. A. (2010). Diagnostic analysis of patients with essential hypertension using association rule mining. *Healthcare Informatics Research,16*(2), 77–81.
10. Agrawal, R., & Srikant, R. (1994). Fast algorithms for mining association rules in large databases. In: J. B. Bocca, M. Jarke, C. Zaniolo (Eds.), *VLDB'94, Proceedings of 20th international conference on very large data bases*, September 12–15, 1994, Santiago de Chile, Chile (pp. 487–499). Morgan Kaufmann.
11. Nahar, J., Imam, T., Tickle, K. S., & Chen, Y. P. (2013). Association rule mining to detect factors which contribute to heart disease in males and females. *Expert Systems with Applications, 40*(4), 1086–1093.
12. Palaniappan, S., & Awang, R. (2008). Intelligent heart disease prediction system using data mining techniques. In: *The 6th ACS/IEEE international conference on computer systems and applications, AICCSA 2008, Doha, Qatar*, March 31-April 4, 2008 (pp. 108–115). IEEE Computer Society.
13. Qamar, N., Yang, Y., Nádas, A., & Liu, Z. (2016). Querying medical datasets while preserving privacy. In: E. M. Shakshuki (Ed.), *The 7th international conference on emerging ubiquitous systems and pervasive networks (EUSPN 2016)/The 6th international conference on current and future trends of information and communication technologies in healthcare (ICTH-2016)/affiliated workshops, September 19–22, 2016, London*, *Procedia Computer Science* (Vol. 98, pp. 324–331). Elsevier.
14. Liu, X., Deng, R. H., Yang, Y., Tran, N. H., & Zhong, S. (2018). Hybrid privacy-preserving clinical decision support system in fog-cloud computing. *Future Generation Computer Systems, 78,* 825–837.
15. Baroni, L., Salles, R., & S.S.E.A. (2020). An analysis of malaria in the Brazilian legal amazon using divergent association rules. *Journal of Biomedical Informatics, 108,* 103512.
16. Bostrom, A. C., Schafer, P., & K. D. E. A. (2006). Electronic health record. Cin Computers Informatics. *Nursing,24*(1), 44–52.
17. Jensen, P. B., & Brunak, L. J. J. (2012). Mining electronic health records: Towards better research applications and clinical care. *Nature Reviews Genetics, 13,* 395–405.
18. Gkoulalas-Divanis, A., Loukides, G., & Sun, J. (2014). Publishing data from electronic health records while preserving privacy: A survey of algorithms. *Journal of Biomedical Informatics, 50,* 4–19.
19. Domadiya, N., & Rao, U. P. (2018). Privacy-preserving association rule mining for horizontally partitioned healthcare data: a case study on the heart diseases. *Sadhana, 43*(8), 127.
20. Nikunj Domadiya, U. P. R. (2019). Privacy preserving distributed association rule mining approach on vertically partitioned healthcare data. *Procedia Computer Science, 148,* 303–312.
21. Yigzaw, K. Y., Budrionis, A., Marco-Ruiz, L., Henriksen, T. D., Halvorsen, P. A., & Bellika, J. G. (2020). Privacy-preserving architecture for providing feedback to clinicians on their clinical performance. *BMC Medical Informatics Decision Making, 20*(1), 116.
22. Nanavati, N. R., & P.L., Jinwala, D.C. (2014). Analysis and evaluation of schemes for secure sum in collaborative frequent itemset mining across horizontally partitioned data. The. *Journal of Engineering, 2014,* 1–10.
23. Diffie, W., & Hellman, M. E. (1976). New directions in cryptography. *IEEE Transactions on Information Theory, 22*(6), 644–654.
24. Paillier, P. (1999). Public-key cryptosystems based on composite degree residuosity classes. In: J. Stern (Ed.), *Advances in cryptology-EUROCRYPT '99, international conference on the theory and application of cryptographic techniques, Prague, Czech Republic, May 2–6, 1999, proceeding*, *Lecture Notes in Computer Science* (Vol. 1592, pp. 223–238). Springer
25. Shoup, V. (2017). The number theory library (ntl). http://www.shoup.net
26. Cleveland heart disease data details (2016). http://archive.ics.uci.edu/ml/machine-learning-databases/heart-disease/heart-disease.names

Wenju Xu received the B.S. and M.S. degree in mathematics from Henan Normal University in 2014, 2017 respectively. She is currently pursuing the Ph.D.'s degree in Xidian University. Her main research interests include public key cryptography and fully homomorphic encryption.

Qingqing Zhao received the M.S. degree in cryptography from the School of Telecommunications Engineering, Xidian University. Her main research interests include privacy-preserving and public key cryptography.

Yu Zhan received his Ph.D. degree in cryptography from Xidian University in 2021, and received the B.S. degree from Chang'an University in 2015. He is currently with the School of Cyber Engineering, Xidian University. His main research interests include public key cryptography and cryptanalysis.

Baocang Wang received the B.S. and M.S. degrees in mathematics, and the Ph.D. degree in cryptography from Xidian University in 2006, 2004, and 2001, respectively. He is currently a professor with the School of Telecommunications Engineering in Xidian University, and Cryptographic Research Center in Xidian University respectively. His main research interests include public key cryptography, wireless network security, fully homomorphic encryption and data mining.

Yupu Hu received the M.S. degree in mathematics and the Ph.D. degree in cryptology from Xidian University, Xi'an, China, in 1987 and 1999, respectively, where he is currently a Professor with the Telecommunication College. He is also serving as one of the directors of the Chinese Association for Cryptologic Research. His major research interests include cryptology, including stream ciphers, block ciphers, and public key ciphers.