This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2020.3026631, IEEE Transactions on Dependable and Secure Computing

Achieve Efficient and Privacy-preserving Disease Risk Assessment over Multi-outsourced Vertical Datasets

Fengwei Wang, *Student Member, IEEE,* Hui Zhu✉, *Senior Member, IEEE,* Rongxing Lu, *Senior Member, IEEE,* Yandong Zheng, Hui Li, *Member, IEEE,*

Abstract—It is believed that online disease risk assessment system has great potential to alleviate the medical treatment problems for the future smart city and communities, as it can excavate disease risk factors from a large number of patient features, provide diagnostic references for doctors, and save medical treatment time for patients. However, the flourish of online disease risk assessment service still faces severe challenges including information privacy and security. In this paper, based on the naïve bayesian classification, we propose an efficient and privacy-preserving disease risk assessment scheme over multi-outsourced vertical datasets, named CARER. With CARER, the e-healthcare provider can securely train a disease risk predication model over vertically distributed medical data from multiple medical centers (i.e., hospitals), and provide privacy-preserving disease risk predication services for users (i.e., patients and doctors). During the model training and disease risk prediction phases, all sensitive data are operated over ciphertexts without decryption. As a result, the private information of medical centers, e-healthcare provider, and users can be well protected. Detailed security analysis shows that CARER can resist various known security threats. In addition, we evaluate the performance of CARER with real medical datasets, and the results demonstrate that CARER is efficient.

Index Terms—Disease risk assessment, privacy-preserving, secure data training, naïve bayesian classification.

1 Introduction

UNDER big data-driven society, a large amount of data is already being collected in an e-healthcare system to generate insights on disease prediction and to improve patient care [1], [2]. As one of the most popular applications in e-healthcare, online disease risk assessment is revolutioning traditional medical, as it can detect a risk condition before it becomes an illness or disorder, and the cost of intervention is far less than the eventual cost of treatment [3], [4]. In general, disease risk assessment mainly consists of two phases, i.e., model training and disease risk prediction. As shown in Fig. 1, in the model training phase, the e-healthcare provider collects and aggregates local medical data from multiple medical centers, and trains a disease risk prediction model based on machine learning algorithms [5], [6]. While in the disease risk prediction phase, the e-healthcare provider can offer online disease risk prediction services for users with the trained prediction model, which will significantly improve the medical treatment efficiency and the quality of people's life.

Unfortunately, owing to the sensitivity of medical information, the flourish of online disease risk assessment is still confronted with severe hassles including data privacy and security [7], [8], [9]. Firstly, local medical data generally contain massive patients' treatment records and statistical data of medical centers, which may disclose patients' individual information and medical centers' clinical treatment programs when outsourcing them to the e-healthcare provider. Secondly, the trained disease risk prediction model is commonly regarded as valuable business assets. Leakage of the prediction model may directly result in an economic loss of the e-healthcare provider. Thirdly, users' disease risk query requests and corresponding query results are also high sensitive, since they may reveal users' private information, such as health conditions, illness, and medication situation during the disease risk prediction. Therefore, in online disease risk assessment, these sensitive data of

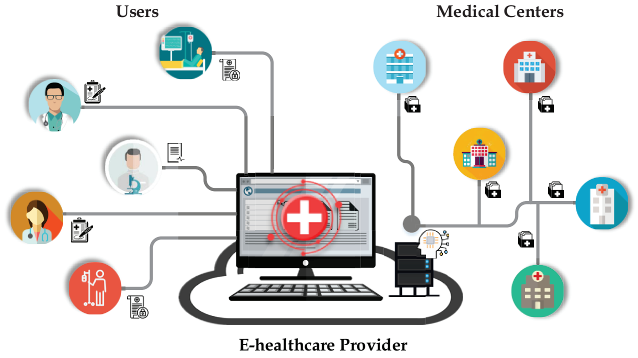


Fig. 1. Conceptual architecture of online disease risk assessment service.

- *F. Wang, H. Zhu, and H. Li are with the State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an, 710071, China (e-mail:xdwangfengwei@gmail.com; zhuhui@xidian.edu.cn; lihui@mail.xidian.edu.cn).*
- *F. Wang, R. Lu and Y. Zheng are with Faculty of Computer Science, University of New Brunswick, New Brunswick, Canada (e-mail:rlu1@unb.ca; yzheng8@unb.ca).*

medical centers, e-healthcare provider and users cannot be leaked to each other.

To address the above-mentioned challenges, plenty of privacy-preserving medical data processing schemes have been proposed, which mainly rely on homomorphic encryption [10], [11] and secure multi-party computation (SMC) technique [12], [13]. Specifically, homomorphic encryption supports arithmetical operations over ciphertexts, which can well protect sensitive medical data during disease risk assessment. However, most homomorphic encryption-based schemes bring heavy computation overhead since they contain massive time-consuming operations such as bilinear pairing. SMC-based schemes can achieve privacy-preserving model training or disease risk prediction, but most of them require multiple interactions to complete a specific operation over ciphertexts, which will bring massive extra communication overhead in distributed scenarios. Moreover, few of existing privacy-preserving medical data processing schemes work on vertically distributed datasets.

In this paper, based on the naïve bayesian classification, we propose an efficient and privacy-preserving online disease risk assessment scheme over multi-outsourced vertical datasets, named CARER. With CARER, the e-healthcare provider can securely train a disease risk predication model over vertically distributed medical data from multiple medical centers, and provide privacy-preserving disease risk prediction services for users. During the process, the sensitive information of medical centers, e-healthcare provider, and users can be well protected. Specifically, the main contributions of this paper are threefold.

- *First*, CARER achieves disease risk prediction model training over vertically distributed data and the trained model can be dynamically updated. In CARER, even if medical centers collect different attributes from instances, the disease risk prediction model can also be effectively trained by the e-healthcare provider. In addition, a model updating strategy is designed, which allows medical centers to upload their fresh collected medical data for updating the prediction model regularly.
- *Second*, CARER is privacy-preserving in both model training and disease risk predication. In CARER, we propose a modified Paillier cryptosystem, with which the prediction model can be securely trained without disclosing sensitive data of medical centers. Besides, random masking technique is applied in disease risk prediction, which preserves the disease risk prediction model of the e-healthcare provider and disease risk query requests/results of users. Therefore, all sensitive data are protected in CARER.
- *Third*, CARER is computationally and communication efficient in both model training and disease risk predication. During the model training phase, the encryption times and communication overhead are greatly reduced with *data pre-processing* in medical centers. Besides, the high-efficiency of disease risk prediction phase is also ensured through simplifying arithmetical operations of naïve bayesian classification. The evaluation with real medical datasets shows that CARER is efficient and can be implemented in the real environment.

The remainder of this paper is organized as follows. In section 2, we formalize the models and identify our design goal. In section 3, we review the Paillier cryptosystem and disease risk prediction model with naïve bayesian classification as preliminaries. Then, we propose our CARER in section 4, followed by the security analysis and performance evaluation are presented in section 5 and section 6, respectively. We also review some related works in section 7. Finally, we draw our conclusions in section 8.

2 Models, Security Requirements and Design Goal

In this section, we first formalize the system model, threaten model, and security requirements. Then, we identify our design goal.

2.1 System Model

In our system model, we mainly focus on how to provide privacy-preserving medical data training and disease risk prediction for online disease risk assessment system. Each medical center/user is equipped with a PC/mobile device, which can connect with the e-healthcare provider. Specifically, the system consists of four parts: 1) trusted authority (TA); 2) medical centers (MCs); 3) e-healthcare provider (EP) and 4) users. As shown in Fig. 2.

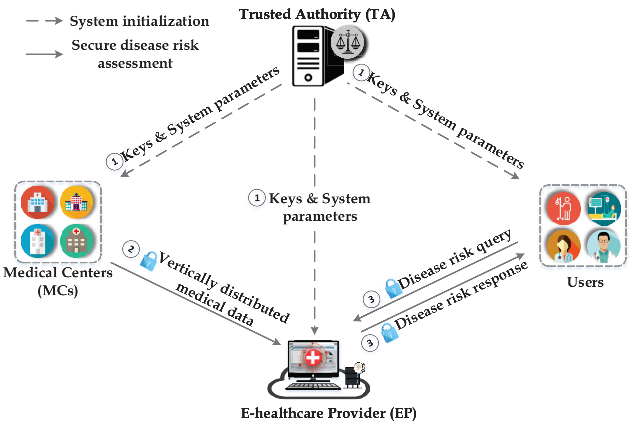

System initialization
Secure disease risk assessment
Trusted Authority (TA)
① Keys & System parameters
① Keys & System parameters
① Keys & System parameters
Medical Centers (MCs)
Users
② Vertically distributed medical data
③ Disease risk query
③ Disease risk response
E-healthcare Provider (EP)

Fig. 2. System model under considered.

- TA is a trusted authority (i.e., a government organization), which bootstraps the whole system through generating system parameters, and distributing keys for medical centers, e-healthcare provider, and users.
- MCs $= \{\mathrm{MC}_1, \cdots, \mathrm{MC}_m\}$ is a set of m medical centers. In our system, each $\mathrm{MC}_i \in$ MCs owns its local medical dataset. Moreover, each MC_i will execute the pre-processing and encryption operations to generate its encrypted local training data, and outsource the ciphertexts to the e-healthcare provider. Note that, different medical centers may collect different attributes in our scheme.

- EP is the e-healthcare provider, which is an online healthcare organization offering disease risk assessment. EP is responsible for aggregating the encrypted local training data from multiple medical centers, training the disease risk prediction model, and offering privacy-preserving disease risk prediction services for users.
- Users are patients or doctors in the e-healthcare system, which are represented as $\{\mathrm{U}_1, \cdots, \mathrm{U}_n\}$. Each $\mathrm{U}_i \in \{\mathrm{U}_1, \cdots, \mathrm{U}_n\}$ owns a symptom vector collected by medical sensors, with which the user can compute her/his encrypted disease risk query request, and further access disease risk prediction services from EP.

2.2 Threaten Model and Security Requirements

In our threaten model, we consider that MCs, EP and users are *honest-but-curious* [14]. Specifically, MCs and EP honestly execute the operations during the model training process, but for commercial interests, a MC wants to obtain other MCs' local training data and EP's prediction model. Besides, EP is also greedy about each MC's local training data. Moreover, during the disease risk prediction process, users and EP execute the protocol strictly, but EP attempts to analyze the accurate symptom vectors and disease risk query results of users. Besides, users are also curious about the disease risk prediction model of EP. Note that there may be some other attacks (i.e., poisoning attacks and denial of service) in an e-healthcare system. Since our target is to protect sensitive data of MCs, EP, and users in disease risk assessment, these attacks are currently out of scope of this paper and will be considered in future work. Considering the above security issues, the following security requirements should be satisfied.

- *Confidentiality:* Ensuring the security of MCs' local training data and EP's prediction model. In general, both of the local training data and the trained disease risk prediction model are regarded as private properties of enterprises. Therefore, during the disease risk assessment, the local training data of MCs and the prediction model of EP cannot be revealed.
- *Privacy:* Protecting users' symptom vectors and disease query results from EP. Since the symptom vector and disease risk query results can reflect a user's private information, during the disease risk assessment, it should be ensured that EP learns no information about users' accurate symptom vectors and the final disease risk query results.

2.3 Design Goal

Under the above-mentioned system model and security requirements, our design goal is to develop an efficient and privacy-preserving disease risk assessment scheme over multi-outsourced vertically distributed medical data. Specifically, the following three objectives should be achieved.

- *Achieve vertically distributed medical data training and updating.* In practice, medical data are distributively stored in multiple medical centers, and different medical centers may collect different attributes from instances. Therefore, CARER should support data aggregation and training over vertically distributed data. Moreover, data updating is also required for renewing the prediction model regularly.
- *Guarantee security and privacy preservation.* Medical data privacy and security is always a vexing problem lying ahead the disease risk assessment system. Once the sensitive medical data of MCs, EP, and users are disclosed, it may lead to serious consequences. Therefore, protecting the local training data of MCs, disease risk prediction model of EP, and symptom vectors/disease risk query results of users should be guaranteed.
- *Low computation and communication overhead.* Although the computational capabilities of servers and mobile devices are increasing rapidly, and the communication between MCs and EP is equipped with high-bandwidth with low-delay. However, it is still difficult for EP to handle huge amounts of medical data. Meanwhile, the batteries of users' mobile devices are still limited. Considering the above factors, the proposed scheme should accomplish high-efficiency in terms of computation and communication.

3 Preliminaries

In this section, we review Paillier cryptosystem, naïve bayesian classification, and introduce the disease risk prediction with naïve bayesian classification, which will serve as the basis of our scheme.

3.1 Paillier Cryptosystem

We apply Paillier cryptosystem [15] as a building block of CARER, which is a widely used public key cryptography with additive homomorphism. Here, we briefly review the Paillier cryptosystem as follows.

- *Key Generation:* Choose the security parameter κ and two big primes $|p| = |q| = \kappa$, compute $N = p \cdot q$ and $\lambda = lcm(p-1, q-1)$. Then, select a random number $g \in \mathbb{Z}_{N^2}^*$ satisfying $gcd(L(g^\lambda \bmod N^2), N) = 1$, where $L(x) = (x-1)/N$. Moreover, compute $\mu = (L(g^\lambda \bmod N^2))^{-1} \bmod N$. Then, the public key pk is (N, g), and the corresponding secret key sk is (μ, λ).
- *Encryption:* Given a message $m \in \mathbb{Z}_N$, the ciphertext can be computed with the public key pk as $c = E_{pk}(m) = g^m \cdot r^N \bmod N^2$, where r is a random number in $\mathbb{Z}_N^*$.
- *Decryption:* Given a ciphertext $c \in \mathbb{Z}_{N^2}^*$, the corresponding plaintext can be retrieved with the secret key sk through computing $m = D_{sk}(c) = L(c^\lambda \bmod N^2) \cdot \mu \bmod N$.

The additive homomorphism of Paillier can be described as: for two arbitrary ciphertexts $c_1 = E_{pk}(m_1)$ and $c_2 = E_{pk}(m_2)$, we have $c_1 \cdot c_2 = E_{pk}(m_1) \cdot E_{pk}(m_2) = g^{m_1+m_2}(r_1 r_2)^N = E_{pk}(m_1 + m_2)$.

3.2 Naïve Bayesian Classification

Naïve bayesian classification [16] is a very concise and useful classifier, which can be used to calculate the posterior probability of an unconfirmend instance belonging to a certain class. Assume that there are v classes denoted as $\{y_1, y_2, \cdots, y_v\}$, and an instance is described as a attribute vector $\mathbf{X} = (x_1, x_2, \cdots, x_u)$. The posterior probability of $\mathbf{X}$ belonging to each class $y_j \in \{y_1, y_2, \cdots, y_v\}$ can be computed with Bayes theorem

$$\Pr(y_j|\mathbf{X}) = \frac{\Pr(\mathbf{X}|y_j)\Pr(y_j)}{\Pr(\mathbf{X})},$$

where $i = 1, 2, \cdots, f$, $\Pr(\mathbf{X}|y_j)$ is the conditional probability of $\mathbf{X}$, $\Pr(y_j)$ and $\Pr(\mathbf{X})$ are the prior probabilities of y_j and $\mathbf{X}$, respectively. Since $\Pr(\mathbf{X})$ is the same for all classes, only $\Pr(\mathbf{X}|y_j)\Pr(y_j)$ needs to be calculated. Moreover, it can be inferred that $\mathbf{X}$ lies in the class $y_{j'}$, if and only if $\Pr(y_{j'}|\mathbf{X})$ is the maximum value in $\Pr(y_j|\mathbf{X})$, where $j = 1, 2, \cdots, v$.

The naïve bayesian classification assumes that all attributes of $\mathbf{X}$ are conditionally independent of each other. Therefore, $\Pr(\mathbf{X}|y_j)\Pr(y_j)$ can be calculated as

$$\Pr(\mathbf{X}|y_j) \cdot \Pr(y_j) = \prod\nolimits_{i=1}^{u} \Pr(x_i|y_j) \cdot \Pr(y_j).$$

Obviously, $\Pr(x_i|y_j)$, where $i = 1, 2, \cdots, u$, can be easily obtained from the training dataset.

3.3 Disease Risk Prediction with Naïve Bayesian Classification

Naïve bayesian classification can be used in predicting the risk of suffering diseases for e-healthcare system [11], [17], [18]. In the following, we introduce the disease risk prediction with naïve bayesian classification.

Given the training dataset containing a large number of confirmed clinical instances, where each instance consists of two binary vectors: $\mathbf{X} = (x_1, x_2, \cdots, x_u) \in \{0,1\}^u$ is the symptom vector with u attributes, and $\mathbf{Y} = (y_1, y_2, \cdots, y_v) \in \{0,1\}^v$ is the disease set with v disease classes. Specifically, $x_s = 1$ means that the instance has the symptom x_s, and $x_s = 0$ otherwise; $y_t = 1$ means that the instance suffers from the disease y_t, and $y_t = 0$ otherwise. With the training dataset, a naïve bayesian classifier can be trained. For example, $\Pr(x_s = 1|y_t = 1)$ can be estimated as $\Pr(x_s = 1|Y_t = 1) = Nz_{ts}/Ny_t$, where Nz_{ts} is the number of instances who have the symptom x_s and suffer from the disease y_t, Ny_t is the number of instances who suffer from the disease y_t. Similarly, the probabilities $\Pr(x_s = 0|y_t = 1)$, $\Pr(y_t = 1)$, $\Pr(x_s = 1|y_t = 0)$, $\Pr(x_s = 0|y_t = 0)$, and $\Pr(y_t = 0)$, where $s = 1, 2, \cdots, u$ and $t = 1, 2, \cdots, v$, can also be easily obtained.

To predict the disease risk of a user with symptom vector $\mathbf{X'} = (x'_1, x'_2, \cdots, x'_u)$. the posterior probability $P(y_t|\mathbf{X'})$ can be estimated as follows.

$$\Pr(y_t = 1|\mathbf{X'}) = \frac{\prod_{s=1}^{u} \Pr(x_s = x'_s|y_t = 1) \cdot \Pr(y_t = 1)}{\Pr(\mathbf{X'})}$$

$$\Pr(y_t = 0|\mathbf{X'}) = \frac{\prod_{s=1}^{u} \Pr(x_s = x'_s|y_t = 0) \cdot \Pr(y_t = 0)}{\Pr(\mathbf{X'})},$$

where $t = 1, 2, \cdots, v$. Finally, if $\Pr(y_t = 1|\mathbf{X'}) > \Pr(y_t = 0|\mathbf{X'})$, it can be inferred that the risk of suffering from disease y_t is greater than not suffering from disease y_t. Moreover, through sorting the posterior probabilities of all diseases in $\mathbf{Y}$, the top-k disease risk list can also be obtianed.

4 Proposed Privacy-preserving Scheme

In this section, we present our CARER scheme, which mainly consists of four phases: 1) *system initialization;* 2) *medical data preprocessing and encryption;* 3) *secure data aggregation and training* and 4) *privacy-preserving disease risk prediction*. The overview of CARER is described in Fig. 3. At first, each MC$_i$ executes data preprocessing and encryption operations on its local medical dataset to generate encrypted local training data, which is further used to train the disease risk prediction model by EP. Then, users encrypt their symptom vectors for generating disease risk query requests, and EP will provide the privacy-preserving disease risk prediction service for users. To describe CARER clearer, we give the description of used notations in Table 1.

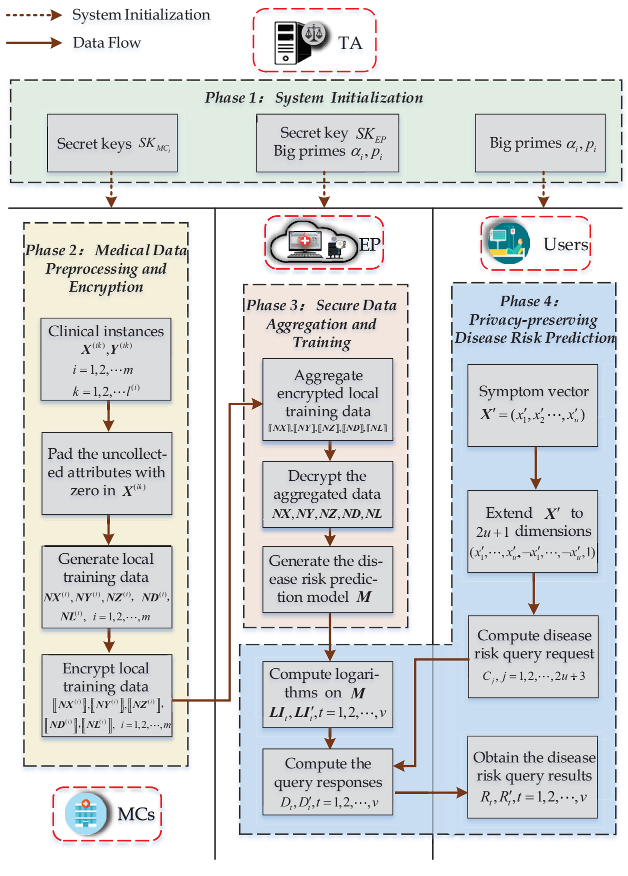

System Initialization
Data Flow
TA
Phase 1: System Initialization
Secret keys SK_{MC_i}
Secret key SK_{EP} Big primes α_i, p_i
Big primes α_i, p_i
EP
Users
Phase 2: Medical Data Preprocessing and Encryption
Clinical instances $X^{(ik)}, Y^{(ik)}$ $i = 1,2,\cdots m$ $k = 1,2,\cdots l^{(i)}$
Pad the uncollected attributes with zero in $X^{(ik)}$
Generate local training data $NX^{(i)}, NY^{(i)}, NZ^{(i)}, ND^{(i)}, NL^{(i)}$, $i = 1,2,\cdots,m$
Encrypt local training data $[\![NX^{(i)}]\!], [\![NY^{(i)}]\!], [\![NZ^{(i)}]\!], [\![ND^{(i)}]\!], [\![NL^{(i)}]\!]$, $i = 1,2,\cdots,m$
MCs
Phase 3: Secure Data Aggregation and Training
Aggregate encrypted local training data $[\![NX]\!], [\![NY]\!], [\![NZ]\!], [\![ND]\!], [\![NL]\!]$
Decrypt the aggregated data NX, NY, NZ, ND, NL
Generate the disease risk prediction model M
Compute logarithms on M $LI_t, LI'_t, t = 1,2,\cdots,v$
Compute the query responses $D_t, D'_t, t = 1,2,\cdots,v$
Phase 4: Privacy-preserving Disease Risk Prediction
Symptom vector $X' = (x'_1, x'_2 \cdots, x'_u)$
Extend X' to $2u+1$ dimensions $(x'_1, \cdots, x'_u, -x'_1, \cdots, -x'_u, 1)$
Compute disease risk query request $C_j, j = 1,2,\cdots,2u+3$
Obtain the disease risk query results $R_t, R'_t, t = 1,2,\cdots,v$

Fig. 3. Overview of CARER.

4.1 System Initialization

During the system initialization phase, TA generates the system parameters, and distributes keys for MCs, EP, and users, respectively.

TA first chooses security parameters $\kappa, k_1, k_2, k_3, k_4$, s.t. $k_4 + 2k_2 < k_1$, $k_2 + k_3 < k_1$, $k_3 + k_4 < k_2$, and executes $Gen(\kappa)$ to generate parameters of Paillier cryptosystem,

TABLE 1
Definition of Notations in CARER

Notation	Definition
$\kappa, k_1, k_2, k_3, k_4$	Security parameters.
$\lvert x \rvert$	The bit length of x.
$[\![x]\!]$	The ciphertext of x with the modified Paillier encryption.
$\mathbf{X}^{(ik)}$	The symptom vector of k-th instance in i-th MC$_i$.
$\mathbf{Y}^{(ik)}$	The disease set of k-th instance in i-th MC$_i$.
SK_{MC_i}	The secret key of MC$_i$.
SK_{EP}	The secret key of EP.
SK_{U_i}	The secret key of U$_i$.
α_i, p_i	Big primes of U$_i$ used to generate query request.
$\mathbf{V}^{(i)}$	The local training data of MC$_i$, $\mathbf{V} = \mathbf{NX}$, $\mathbf{NY}$, $\mathbf{NZ}$, $\mathbf{ND}$, $\mathbf{NL}$.
$[\![\mathbf{V}^{(i)}]\!]$	Encrypted local traning data of MC$_i$.
$[\![\mathbf{V}]\!]$	The aggregated ciphertext of $[\![\mathbf{V}^{(i)}]\!]$.
$\mathbf{M}$	Generated disease risk prediction model.
$\mathbf{X}'$	The symptom vector of U$_i$.
C_j	Disease query request, $j = 1, 2, \cdots, 2u+3$.
$\mathbf{LI}_t, \mathbf{LI}'_t$	Logarithm values of probabilities in disease risk prediction model, $t = 1, 2, \cdots, v$.
D_t, D'_t	Disease risk query responses, $t = 1, 2, \cdots, v$.
R_t, R'_t	The disease risk query results, $t = 1, 2, \cdots, v$.

which includes the secret key $SK_p = (\mu, \lambda)$ and the public key $PK_p = (N, g)$. Then, TA selects a large random number γ satisfying $|\gamma| < \frac{\kappa}{2}$, and computes $h = g^\gamma \bmod N^2$. Note that MCs may collect different attributes of instances, a list $\mathcal{L}$ covering all collected attributes needs to be generated by TA. Finally, TA publishes the public parameters $< \kappa, k_1, k_2, k_3, k_4, N, g, h, \mathcal{L} >$.

For MCs, TA splits N to m random numbers $\{n_1, n_2, \cdots, n_m\}$, s.t. $\sum_{i=1}^m n_i = N$, selects a random number s_t in $\mathbb{Z}_N^*$ as the task ID for every medical data aggregation task, and computes the secret keys $SK_{MC_i} = {s_t}^{n_i} \bmod N^2$, which are used to encrypt local training data of MCs.

For EP, TA distributes the secret key $SK_{EP} =< SK_p, \gamma >=< \lambda, \mu, \gamma >$ to EP, which is used to decrypt the data aggregation results.

For each U$_i$, TA chooses two large primes p_i and α_i satisfying $|p_i| = k_1$ and $|\alpha_i| = k_2$, which are used for encrypting symptom vectors. Moreover, TA also sends $< p_i, \alpha_i >$ of each U$_i$ to EP.

4.2 Medical Data Preprocessing and Encryption

In this phase, every MC$_i$ first preprocesses its collected local medical dataset to generate local training data. Moreover, with the secret key SK_{MC_i}, each MC$_i$ encrypts its local training data before sending to EP.

• Step 1. Medical Data Preprocessing

Assume that MC$_i$ owns a local medical dataset $\mathbf{S}^{(i)}$, which contains $l^{(i)}$ confirmed clinical instances. For each instance, MC$_i$ collects w attributes and v diseases, which can be represented as

$$\mathbf{X}^{(ik)} = (x_1^{(ik)}, x_2^{(ik)}, \cdots, x_w^{(ik)}) \in \{0,1\}^w$$
$$\mathbf{Y}^{(ik)} = (y_1^{(ik)}, y_2^{(ik)}, \cdots, y_v^{(ik)}) \in \{0,1\}^v,$$

where $k = 1, 2, \cdots, l^{(i)}$.

At first, MC$_i$ checks if there are uncollected attributes in the List $\mathcal{L}$. If existing, MC$_i$ pads the uncollected attributes with zero for each instance. Assume that the number of attributes in $\mathcal{L}$ is u, thus, $\mathbf{X}^{(ik)}$ is extended to a u-dimensional vector as

$$\mathbf{X}^{(ik)} = (x_1^{(ik)}, x_2^{(ik)}, \cdots, x_u^{(ik)}) \in \{0,1\}^u.$$

Moreover, MC$_i$ generates a vector $\mathbf{E} = (e_1, e_2, \cdots, e_u)$, where $e_s = 1$ if MC$_i$ collects the $s - th$ attribute in $\mathcal{L}$, and $e_s = 0$, otherwise.

Then, for $s = 1, 2, \cdots, u$, $t = 1, 2, \cdots, v$, and $k = 1, 2, \cdots, l^{(i)}$, MC$_i$ computes $Nx_s^{(i)} = \sum_{k=1}^{l^{(i)}} x_s^{(ik)}$, $Ny_t^{(i)} = \sum_{k=1}^{l^{(i)}} y_t^{(ik)}$, $Nz_{ts}^{(i)} = \sum_{k=1}^{l^{(i)}} x_s^{(ik)} \cdot y_t^{(ik)}$, $Nd_{ts}^{(i)} = Ny_t^{(i)} \cdot e_s$, and $Nl_s^{(i)} = l^{(i)} \cdot e_s$. After this, the local training data

$$\mathbf{NX}^{(i)} = (Nx_1^{(i)}, Nx_2^{(i)}, \cdots, Nx_u^{(i)})$$
$$\mathbf{NY}^{(i)} = (Ny_1^{(i)}, Ny_2^{(i)}, \cdots, Ny_v^{(i)})$$
$$\mathbf{NZ}^{(i)} = (Nz_{11}^{(i)}, \cdots, Nz_{ts}^{(i)}, \cdots, Nz_{vu}^{(i)})$$
$$\mathbf{ND}^{(i)} = (Nd_{11}^{(i)}, \cdots, Nd_{ts}^{(i)}, \cdots, Nd_{vu}^{(i)})$$
$$\mathbf{NL}^{(i)} = (Nl_1^{(i)}, Nl_2^{(i)}, \cdots, Nl_u^{(i)}, l^{(i)})$$

can be obtained. In detail, for $s = 1, 2, \cdots, u$ and $t = 1, 2, \cdots, v$, $\mathbf{NX}^{(i)}$, $\mathbf{NY}^{(i)}$, and $\mathbf{NZ}^{(i)}$ represent the number of instances who have symptom x_s, suffer from disease y_t, and have symptom x_s while suffering from disease y_t, respectively. Besides, in order to construct prediction model with vertically distributed datasets, vectors $\mathbf{ND}^{(i)}$ and $\mathbf{NL}^{(i)}$ should be computed in our scheme. Specifically, $\mathbf{ND}^{(i)}$ is derived from $\mathbf{NY}^{(i)}$ and collected attributes of MC$_i$, while $\mathbf{NL}^{(i)}$ is computed with the total number of instances $l^{(i)}$ and collected attributes of MC$_i$.

• Step 2. Local Training Data Encryption

After generating local training data $< \mathbf{NX}^{(i)}, \mathbf{NY}^{(i)}, \mathbf{NZ}^{(i)}, \mathbf{ND}^{(i)}, \mathbf{NL}^{(i)} >$, for each element $a^{(i)}$ in $< \mathbf{NX}^{(i)}, \mathbf{NY}^{(i)}, \mathbf{NZ}^{(i)}, \mathbf{ND}^{(i)}, \mathbf{NL}^{(i)} >$, MC$_i$ executes encryption operations as follows

$$[\![a^{(i)}]\!] = g^{a^{(i)}} \cdot h^{r_i} \cdot SK_{MC_i} \bmod N^2,$$

where r_i is a random number satisfying $|r_i| < \frac{\kappa}{2}$. After this, MC$_i$ obtains the encrypted local training data

$$< [\![\mathbf{NX}^{(i)}]\!], [\![\mathbf{NY}^{(i)}]\!], [\![\mathbf{NZ}^{(i)}]\!], [\![\mathbf{ND}^{(i)}]\!], [\![\mathbf{NL}^{(i)}]\!] >,$$

and sends it to EP.

4.3 Secure Data Aggregation and Training

In this phase, with the proposed STDA algorithm, EP aggregates the encrypted local training data of MCs, decrypts the aggregated results with the secret key SK_{EP}, and trains the disease risk prediction model.

• Step 1. Secure Data Aggregation

Once the encrypted local training data $< [\![\mathbf{NX}^{(i)}]\!]$, $[\![\mathbf{NY}^{(i)}]\!]$, $[\![\mathbf{NZ}^{(i)}]\!]$, $[\![\mathbf{ND}^{(i)}]\!]$, $[\![\mathbf{NL}^{(i)}]\!] >$, where $i = 1, 2, \cdots, m$, from all MCs are received, EP first executes aggregation operations. Specifically, For each element $[\![a^{(i)}]\!]$ in $< [\![\mathbf{NX}^{(i)}]\!]$, $[\![\mathbf{NY}^{(i)}]\!]$, $[\![\mathbf{NZ}^{(i)}]\!]$, $[\![\mathbf{ND}^{(i)}]\!]$, $[\![\mathbf{NL}^{(i)}]\!] >$, $i = 1, 2, \cdots, m$, EP aggregates the ciphertexts of local training data through computing

$$[\![a]\!] = \prod\nolimits_{i=1}^{m} [\![a^{(i)}]\!] \bmod N^2.$$

After this, EP obtains the encrypted global training data

$$< [\![\mathbf{NX}]\!], [\![\mathbf{NY}]\!], [\![\mathbf{NZ}]\!], [\![\mathbf{ND}]\!], [\![\mathbf{NL}]\!] > .$$

Furthermore, for every $[\![a]\!]$ in $< [\![\mathbf{NX}]\!]$, $[\![\mathbf{NY}]\!]$, $[\![\mathbf{NZ}]\!]$, $[\![\mathbf{ND}]\!]$, $[\![\mathbf{NL}]\!] >$, EP decrypts it with the secret key SK_{EP} as follows

$$a = (L([\![a]\!]^{\lambda} \bmod N^2 \cdot \mu) \bmod N) \bmod \gamma,$$

where $L(x) = \frac{x-1}{N}$. Finally, EP obtains the global training data

$$\begin{aligned} \mathbf{NX} &= (Nx_1, Nx_2, \cdots, Nx_u) \\ \mathbf{NY} &= (Ny_1, Ny_2, \cdots, Ny_v) \\ \mathbf{NZ} &= (Nz_{11}, \cdots, Nz_{ts}, \cdots, Nz_{vu}) \\ \mathbf{ND} &= (Nd_{11}, \cdots, Nd_{ts}, \cdots, Nd_{vu}) \\ \mathbf{NL} &= (Nl_1, Nl_2, \cdots, Nl_u, l). \end{aligned}$$

Algorithm 1: STDA: Secure Training Data Aggregation

```
Input: The local medical datasets of MCs: $\mathbf{S}^{(i)}$, $i = 1, 2, \cdots, m$.
Output: The global training data: NX, NY, NZ, ND and NL.
MC$_i$ preprocesses $\mathbf{S}^{(i)}$ to generate local training data
    $< \mathbf{NX}^{(i)}, \mathbf{NY}^{(i)}, \mathbf{NZ}^{(i)}, \mathbf{ND}^{(i)}, \mathbf{NL}^{(i)} > \leftarrow \mathbf{S}^{(i)}$;
foreach $a^{(i)}$ in $< \mathbf{NX}^{(i)}, \mathbf{NY}^{(i)}, \mathbf{NZ}^{(i)}, \mathbf{ND}^{(i)}, \mathbf{NL}^{(i)} >$ do
    MC$_i$ uses its secret key $SK_{MC_i}$ to encrypt $a^{(i)}$
        $[\![a^{(i)}]\!] \leftarrow (a^{(i)}, SK_{MC_i})$;
end
MC$_i$ obtains the encrypted local training data
    $< [\![\mathbf{NX}^{(i)}]\!], [\![\mathbf{NY}^{(i)}]\!], [\![\mathbf{NZ}^{(i)}]\!], [\![\mathbf{ND}^{(i)}]\!], [\![\mathbf{NL}^{(i)}]\!] >$;
foreach $[\![a^{(i)}]\!]$ in $< [\![\mathbf{NX}^{(i)}]\!], [\![\mathbf{NY}^{(i)}]\!], [\![\mathbf{NZ}^{(i)}]\!], [\![\mathbf{ND}^{(i)}]\!], [\![\mathbf{NL}^{(i)}]\!] >$ and $i = 1, 2, \cdots, m$ do
    EP aggregates the encrypted $[\![a^{(i)}]\!]$ over ciphertexts
        $[\![a]\!] \leftarrow [\![a^{(i)}]\!]$;
    EP uses its secret key $SK_{EP}$ to decrypt $[\![a]\!]$
        $a \leftarrow ([\![a]\!], SK_{EP})$;
end
EP obtains the global training data
    $< \mathbf{NX}, \mathbf{NY}, \mathbf{NZ}, \mathbf{ND}, \mathbf{NL} >$;
```

• Step 2. Disease Risk Prediction Model Training

With the elements in global training data $< \mathbf{NX}, \mathbf{NY}, \mathbf{NZ}, \mathbf{ND}, \mathbf{NL} >$, EP can train the disease risk prediction model $\mathbf{M}$ through computing the following probabilities

$$\begin{gathered} \Pr(x_s = 1 | y_t = 1) = \tfrac{Nz_{ts}}{Nd_{ts}} \\ \Pr(x_s = 1 | y_t = 0) = \tfrac{Nx_s - Nz_{ts}}{Nl_s - Nd_{ts}} \\ \Pr(y_t = 1) = \tfrac{Ny_t}{l}, \Pr(y_t = 0) = 1 - \Pr(y_t = 1) \\ \Pr(x_s = 0 | y_t = 1) = 1 - \Pr(x_s = 1 | y_t = 1) \\ \Pr(x_s = 0 | y_t = 0) = 1 - \Pr(x_s = 1 | y_t = 0). \end{gathered}$$

4.4 Privacy-preserving Disease Risk Prediction

In this phase, with the proposed PDRP algorithm, users encrypt their symptom vectors for generating disease risk query requests, which will be used to compute disease risk query responses over ciphertexts by EP. Finally, users decrypt the query responses for obtaining the final disease risk query results.

• Step 1. Disease Risk Query Generation

Assume that the symptom vector of a user U$_i$ is

$$\mathbf{X}' = (x'_1, x'_2, \cdots, x'_u) \in \{0, 1\}^u.$$

Firstly, U$_i$ inverts each element in $\mathbf{X}'$, and extends $\mathbf{X}'$ to a $(2u + 1)$-dimensional vector such as

$$\begin{aligned} \mathbf{X}' &= (x'_1, x'_2, \cdots, x'_{2u+1}) \\ &= (x'_1, \cdots, x'_u, \neg x'_1, \cdots, \neg x'_u, 1). \end{aligned}$$

Then, U$_i$ sets $x'_{2u+2} = x'_{2u+3} = 0$, chooses a large random number $\sigma \in Z_{p_i}$, and selects $2u + 3$ random numbers c_j, $j = 1, 2, \cdots, 2u + 3$, with $|c_j| = k_3$. Moreover, for each x_j, $j = 1, 2, \cdots, 2u + 3$, U$_i$ computes

$$C_j = \begin{cases} \sigma(x'_j \cdot \alpha_i + c_j) \bmod p_i, x'_j \neq 0 \\ \sigma \cdot c_j \bmod p_i, \qquad\qquad x'_j = 0. \end{cases}$$

Finally, U$_i$ keeps $SK_{U_i} = \sigma^{-1} \bmod p_i$ as her/his secret key, and sends the disease risk query request $< C_1, C_2, \cdots, C_{2u+3} >$ to EP.

• Step 2. Query Response Computation

After receiving query request $< C_1, C_2, \cdots, C_{2u+3} >$ from U$_i$, EP computes the query responses for U$_i$ through the following operations.

Firstly, EP executes the logarithmic calculation on each element in the disease risk prediction model $\mathbf{M}$. Specifically, for $s = 1, 2, \cdots, u$ and $t = 1, 2, \cdots, v$, EP computes

$$\begin{gathered} Lp_{ts} = \log(\Pr(x_s = 1 | y_t = 1)), \\ Ln_{ts} = \log(\Pr(x_s = 0 | y_t = 1)),\ Ld_t = \log(\Pr(y_t = 1)), \\ Lp'_{ts} = \log(\Pr(x_s = 1 | y_t = 0)), \\ Ln'_{ts} = \log(\Pr(x_s = 0 | y_t = 0)),\ Ld'_t = \log(\Pr(y_t = 0)), \end{gathered}$$

where the base of logarithm function can be selected in $(0, 1)$ arbitrarily.

Then, for each disease y_t, EP can generate two $(2u + 1)$-dimensional vectors as

$$\mathbf{LI}_t = (Li_{t1}, Li_{t2}, \cdots, Li_{t(2u+1)})$$

$$
\begin{aligned}
&= (Lp_{t1}, \cdots, Lp_{tu}, Ln_{t1}, \cdots, Ln_{tu}, Ld_t)\\
\mathbf{LI}'_t &= (Li'_{t1}, Li'_{t2}, \cdots, Li'_{t(2u+1)})\\
&= (Lp'_{t1}, \cdots, Lp'_{tu}, Ln'_{t1}, \cdots, Ln'_{tu}, Ld'_t),
\end{aligned}
$$

where $t = 1, 2, \cdots, v$. Moreover, EP computes the disease risk query responses with the query requests of U_i and each vector in $(\mathbf{LI}_1, \cdots, \mathbf{LI}_v, \mathbf{LI}'_1, \cdots, \mathbf{LI}'_v)$.

For the simplification, we take computing the query response with $\mathbf{LI}_t$ as an example. EP first sets $Li_{t(2u+2)} = Li_{t(2u+3)} = 0$, and computes

$$D_{tj} = \begin{cases} Li_{tj} \cdot \alpha_i \cdot C_j \bmod p_i, Li_{tj} \neq 0 \\ r_j \cdot C_j \bmod p_i, \qquad Li_{tj} = 0. \end{cases}$$

where r_j is a random number, with $|r_j| = k_4$. After this, EP computes $D_t = \sum_{j=1}^{2u+3} D_{tj}$.

Finally, EP can obtain $2v$ query responses $< D_1, \cdots, D_v, D'_1, \cdots, D'_v >$, and sends them back to U_i.

- **Step 3. Query Results Reading**

Once query responses $< D_1, \cdots, D_v, D'_1, \cdots, D'_v >$ are received, U_i computes the query results through the following operations.

We also take a D_t in $< D_1, \cdots, D_v, D'_1, \cdots, D'_v >$ as an example. With D_t, U_i computes

$$E_t = \sigma^{-1} \cdot D_t \bmod p_i,\ R_t = \frac{E_t - E_t(\text{mod}\alpha_i^2)}{\alpha_i^2}.$$

Finally, U_i can obtain the disease risk query results from $< R_1, \cdots, R_v, R'_1, \cdots R'_v >$. Specifically, if $R_t < R'_t$, the probability of suffering disease y_t is greater than not suffering disease y_t, and if $R_t > R'_t$, otherwise. Moreover, through sorting $< R_1, R_2, \cdots, R_v >$ from small to large, the risk list of suffering diseases $(y_1, y_2, \cdots, y_v)$ from high to low can be obtained by U_i.

Algorithm 2: PDRP: Privacy-preserving Disease Risk Prediction

```
Input: The symptom vector $\mathbf{X}$ of $\text{U}_i$, and the disease
       risk prediction model $\mathbf{M}$ of EP.
Output: The query results $R_1, \cdots, R_v, R'_1, \cdots, R'_v$.
U_i extends its symptom vector
    $\mathbf{X} \to (x'_1, x'_2, \cdots, x'_{2u+1})$;
U_i sets $x'_{2u+2} = x'_{2u+3} = 0$;
foreach $j = 1, 2, \cdots, 2u+3$ do
    $\text{U}_i$ uses prime numbers $\alpha', p'$ and random
    numbers $c_j$ to generate query requests
        $C_j \leftarrow (x'_j, \alpha_i, p_i, c_j)$;
end
foreach $t = 1, 2, \cdots, v$ do
    EP uses the disease risk prediction model $\mathbf{M}$ to
    compute
        $(\mathbf{LI}_t, \mathbf{LI}'_t) \leftarrow \mathbf{M}$;
    EP uses $(\mathbf{LI}_t, \mathbf{LI}'_t)$ and query request $< C_1, C_2, \cdots$
    $C_{2u+3} >$ to compute
        $D_t \leftarrow (\mathbf{LI}_t, C_1, C_2, \cdots C_{2u+3})$
        $D'_t \leftarrow (\mathbf{LI}'_t, C_1, C_2, \cdots C_{2u+3})$;
    $\text{U}_i$ uses its secret key $SK_{U_i}$ to compute
        $R_t \leftarrow (D_t, SK_{U_i})$
        $R'_t \leftarrow (D'_t, SK_{U_i})$;
end
$\text{U}_i$ obtains the disease risk query results
    $< R_1, \cdots, R_v, R'_1, \cdots, R'_v >$;
```

4.5 Disease Risk Prediction Model Updating

In practice, MCs collect medical data from instances continuously. Therefore, the disease risk prediction model should be updated regularly, which can be achieved with the following operations.

Firstly, TA renews the task ID to s'_t, and updates secret keys for each MC_i through computing $SK_{MC_i} = s_t'^{n_i}$. Then, assume that the new collected dataset of MC_i is represented as $\mathbf{S}'^{(i)}$, via taking $\mathbf{S}'^{(i)}$ as the input of STDA Algorithm, EP and MC can execute STDA to securely aggregate the new collected dataset from m MCs. Finally, EP updates the global training data through adding the aggregated results in it, and further renews the disease risk prediction model with the updated global training data.

4.6 Correctness of the Proposed Scheme

In this section, we prove the correctness of algorithms STDA and PDRP, which support our proposed scheme in basis.

- **Correctness of STDA**

In STDA, EP first aggregates the encrypted local training data from MCs through the following calculations.

$$
\begin{aligned}
[\![a]\!] &= \prod\nolimits_{i=1}^{m} [\![a^{(i)}]\!] \bmod N^2\\
&= \prod\nolimits_{i=1}^{m} g^{a^{(i)}} \cdot h^{r_i} \cdot s_t^{N_i} \bmod N^2\\
&= g^{\sum_{i=1}^{m} a^{(i)}} \cdot h^{\sum_{i=1}^{m} r_i} \cdot s_t^{\sum_{i=1}^{m} n_i} \bmod N^2\\
&= g^{\sum_{i=1}^{m} a^{(i)} + \gamma \cdot \sum_{i=1}^{m} r_i} \cdot s_t^{N} \bmod N^2.
\end{aligned}
$$

Since $|\gamma| = |r_i| < \frac{\kappa}{2}$ and $a^{(i)}$ are not big values, it can be inferred that $\sum_{i=1}^{m} a^{(i)} + \gamma \cdot \sum_{i=1}^{m} r_i$ is in $\mathbb{Z}_N$, which is the plaintext domain of Paillier cryptosystem. Then, EP can correctly decrypt the aggregated results $a = \sum_{i=1}^{m} a^{(i)}$ through computing

$$
\begin{aligned}
a &= (L([\![a]\!]^{\lambda} \bmod N^2 \cdot \mu) \bmod N) \bmod \gamma\\
&= (\sum\nolimits_{i=1}^{m} a^{(i)} + \gamma \cdot \sum\nolimits_{i=1}^{m} r_i) \bmod \gamma\\
&= \sum\nolimits_{i=1}^{m} a^{(i)}.
\end{aligned}
$$

Therefore, it can be seen that the global training data is the summation of the local training data from all MCs. Moreover, uncollected attributes are padded with zero, which makes them no effect on computing the probabilities in $\mathbf{M}$. Finally, we can conclude that EP trains the disease risk prediction model correctly.

- **Correctness of PDRP**

In PDRP, a secure inner product computation protocol [19] is applied to compute the query results R_t, R'_t, $t = 1, 2, \cdots, v$. That is, the query results R_t and R'_t are the inner products of the extended symptom vector $\mathbf{X}'$ of U_i and the vectors $\mathbf{LI}_t, \mathbf{LI}'_t$. Then, we can compute

$$\begin{aligned} R_t &= \mathbf{X}' \cdot \mathbf{LI}_t \\ &= \sum\nolimits_{s=1}^{u} x'_s \cdot \log(\Pr(x_s = 1|y_t = 1)) \\ &\qquad + \sum\nolimits_{s=1}^{u} (1 - x'_s) \cdot \log(\Pr(x_s = 0|y_t = 1)) \\ &\qquad + \log(\Pr(y_t = 1)) \\ &= \sum\nolimits_{s=1}^{u} \log(\Pr(x_s = x'_s|y_t = 1)) + \log(\Pr(y_t = 1)) \\ &= \log(\prod\nolimits_{s=1}^{u} \Pr(x_s = x'_s|y_t = 1) \cdot \Pr(y_t = 1)) \end{aligned}$$

$$\begin{aligned} R'_t &= \mathbf{X}' \cdot \mathbf{LI}'_t \\ &= \sum\nolimits_{s=1}^{u} x'_s \cdot \log(\Pr(x_s = 1|y_t = 0)) \\ &\qquad + \sum\nolimits_{s=1}^{u} (1 - x'_s) \cdot \log(\Pr(x_s = 0|y_t = 0)) \\ &\qquad + \log(\Pr(y_t = 0)) \\ &= \sum\nolimits_{s=1}^{u} \log(\Pr(x_s = x'_s|y_t = 0)) + \log(\Pr(y_t = 0)) \\ &= \log(\prod\nolimits_{s=1}^{u} \Pr(x_s = x'_s|y_t = 0) \cdot \Pr(y_t = 0)). \end{aligned}$$

Since logarithmic is a monotonic function and we set the base in $(0, 1)$, it is obvious that if $D_t < D'_t$, then $\prod_{s=1}^{u} \Pr(x_s = x'_s|y_t = 1) \cdot \Pr(y_t = 1) > \prod_{s=1}^{u} \Pr(x_s = x'_s|y_t = 0) \cdot \Pr(y_t = 0)$, and the risk of suffering the disease y_t is greater than not suffering disease y_t. If $D_t > D'_t$, otherwise. Moreover, if $D_t < D_{t'}$, we can infer that the risk of suffering disease y_t is greater than the risk of suffering disease risk $y_{t'}$. Therefore, users can obtain the disease risk query results correctly.

5 Security Analysis

In this section, we analyze the security of the proposed CARER. Specifically, following the security requirements discussed earlier, our analysis focuses on how to ensure the confidentiality of MCs' local training data and the EP's disease risk prediction model, as well as the privacy of users' symptom vectors/query results.

- **The Confidentiality of MCs' Local Training Data is Achieved.** During the model training phase, the confidentiality of MCs' local training data is achieved based on our encryption algorithm, which is modified from Paillier cryptosystem [15]. In the original Paillier cryptosystem, since the secret key SK_p can decrypt arbitrary ciphertexts encrypted with the public key PK_p, which makes it cannot be used for secure multi-party data aggregation. In our encryption algorithm, we split the parameter N in public key of original Paillier PK_p into m parts $\{n_1, n_2, \cdots, n_m\}$, and generate secret keys $SK_{MC_i} = s_t^{n_i} \bmod N^2$ with the splitted N for m MCs. Therefore, the secret key of original Paillier SK_p cannot decrypt ciphertexts encrypted with SK_{MC_i}. Specifically, in our scheme, each MC_i uses its secret key SK_{MC_i} to encrypt each element $a^{(i)}$ in local training data as $[\![a^{(i)}]\!] = g^{a^{(i)}} \cdot h^{r_i} \cdot SK_{MC_i} \bmod N^2$, which cannot be decrypted by EP even if EP owns the secret key SK_p. However, after aggregating the ciphertexts of MCs, we have the aggregated result $[\![a]\!] = g^{\sum_{i=1}^{m} a^{(i)} + \gamma \cdot \sum_{i=1}^{m} r_i} \cdot s_t^N \bmod N^2$. It can be seen that parameter N in public key of original Paillier PK_p is recovered, and EP is able to decrypt the aggregated result correctly. Therefore, with our modified Paillier cryptosystem, EP can only obtain aggregated information from all MCs, while the local training data of each MC is well protected.

Moreover, for inferring a MC's local training data, EP needs to collude with other $m-1$ MCs, which is impossible in practice. Therefore, collusion attack is resisted in CARER. In addition, for each data aggregation task, a new task id s_t is selected by TA to generate different secret keys for each MC, with which the replay attack is also resisted.

- **The Confidentiality of EP's Disease Risk Prediction Model is Achieved.** During the model training phase, since our proposed STDA algorithm is noninteractive, it can be confirmed that MCs learn no information about EP's sensitive data. Therefore, the disease risk prediction model of EP is kept secret from MCs. During the disease risk prediction phase, we introduce a secure two-party inner product computation protocol to compute the disease risk query responses for users. Specifically, after receiving query request C_j, where $j = 1, 2, \cdots, 2u + 3$, EP combines vectors $\mathbf{LI}_t = (Li_{t1}, Li_{t2}, \cdots, Li_{t(2u+3)})$ and $\mathbf{LI}'_t = (Li'_{t1}, Li'_{t2}, \cdots, Li'_{t(2u+3)})$, which are derived from the disease risk prediction model, with query request C_j. In detail, EP computes $D_{tj} = Li_{tj} \cdot \alpha_i \cdot C_j \bmod p_i$, if $Li_{tj} \neq 0$; $D_{tj} = r_j \cdot C_j \bmod p_i$, if $Li_{tj} = 0$, where $j = 1, 2, \cdots, 2u+3$, and computes query response $D_t = \sum_{j=1}^{2u+3} D_{tj} = \sigma \cdot (\sum_{x'_j \neq 0, Li_{tj} \neq 0} x'_j \cdot Li_{tj} \cdot \alpha_i^2 + \sum_{x'_j \neq 0, Li_{tj} = 0} r_j(x'_j \cdot \alpha_i + c_j) + \sum_{x'_j = 0, Li_{tj} = 0} r_j \cdot c_j) \bmod p_i$ for users. Note that $Li_{t(2u+2)} = Li_{t(2u+3)} = 0$ ensures that at least two random numbers are included in D_t, which can prevent users from guessing vectors $\mathbf{LI}_t$ and $\mathbf{LI}'_t$ of EP. Therefore, the disease risk prediction model can be well protected from users.

Furthermore, in our proposed scheme, each U_i first extends its symptom vector to a $(2u+3)$-dimensional vector, in which the numbers of 1 and 0 are $u+1$ and $u+2$, respectively. Therefore, without modifying the extended vector containing only one 1 maliciously, it is impossible for U_i to obtain accurate vectors $\mathbf{LI}_t$ and $\mathbf{LI}'_t$ through inferring their values in a specific dimension. Therefore, the confidentiality of disease risk prediction model is achieved.

- **The Privacy of Users' Symptom Vectors and Query Results is Guaranteed.** For accessing disease risk prediction services without disclosing sensitive data, each U_i first extends her/his symptom vector into $(2u+3)$-dimensional, and encrypts the extended vector $\mathbf{X}' = (x'_1, x'_2, \cdots, x'_{2u+3})$ through computing $C_j = \sigma(x'_j \cdot \alpha_i + c_j) \bmod p_i$, if $x'_j \neq 0$; $C_j = \sigma \cdot c_j \bmod p_i$, if $x'_j = 0$, where $j = 1, 2, \cdots, 2u+3$. We can see that each x'_j is one-time masked with random number c_j, therefore, U_i can ensure that each x_j is privacy-preserving against EP. Besides, since parameter σ is only known by U_i, EP cannot retrieve the disease risk query results R_t and R'_t. Then, both of the symptom vector and query results of U_i can be well protected.

In addition, even if an attacker can eavesdrop communications between EP and users, and obtains all packets from users and EP, it cannot obtain any useful information since there are massive random numbers contained in query requests and responses. Then, the privacy of users' symptom vectors and query results is Guaranteed.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2020.3026631, IEEE Transactions on Dependable and Secure Computing

6 Performance Evaluation

In this section, we analyze and test the performance of the proposed CARER in terms of computation cost and communication overhead, and make the comparison with the EPDP [17].

6.1 Evaluation Environment

In order to measure the integrated performance, we conduct the experiment in Java running on the PC with one 2.2-GHz Intel Core i7, 16-GB memory, and Windows 10 system. Moreover, For CARER, we set the security parameters as $\kappa = 1024$, $k_1 = 512$, $k_2 = 200$, $k_3 = 32$, and $k_4 = 32$. For EPDP, we also set the security parameter of homomorphic encryption $\kappa = 1024$, choose AES-256 and HMAC-SHA1 as the symmetric encryption algorithm and keyed cryptographic hash function, respectively. Besides, we also set the size of bloom filter in EPDP as 2^{27}. Similar to [17], we consider two datasets. One real dataset is UCI machine learning repository called Breast Cancer Wisconsin (Original) dataset (BCW) [20]. We use the BCW dataset to test the prediction accuracy for the proposed CARER. Besides, we also use the synthetic dataset to test all factors affecting the performance of CARER, and make the comparison with EPDP. The details of the two datasets are described as follows.

- Real dataset (BCW): A medical dataset periodically collected by a medical expert, which contains 699 instances used to execute the pre-diagnosis of suffering breast cancer. Specifically, each instance contains nine symptom attributes (each attribute ranges from 1 to 10) and a target variable (2 for benign, 4 for malignant).
- Synthetic dataset: In order to test the integrated performance of the proposed CARER, we generate a synthetic dataset randomly, which consists of 3000 instances. Each instance contains 20 attributes and 10 targets variables, and the value of each element is randomly picked either 0 or 1.

6.2 Accuracy Evaluation

For verifying the effectiveness of CARER, we test the prediction accuracy of CARER with BCW dataset, and make a comparison with the case without privacy-preserving mechanism. We first choose 3 subsets from BCW dataset for simulating the local medical datasets of 3 MCs. Specifically, MC$_1$ owns 150 instances with third to ninth attributes, MC$_2$ owns 150 instances with first to seventh attributes, and MC$_3$ owns 183 instances with all nine attributes. Moreover, Since the value of attributes in BCW dataset ranges from 1 to 10, each attribute should be extended to 10-dimension for normalizing the dataset into binary. Then, we use the 3 local medical datasets to train two disease risk prediction models (one is trained over plaintext, while the other is trained with CARER), and test the prediction accuracy of them with the other 200 instances in BCW dataset. Table 2 shows the test results, it can be seen that our proposed scheme can provide high-accuracy disease risk prediction results, while the accuracy is not affected by the privacy-preserving mechanism.

TABLE 2
Accuracy Evaluation of CARER compared with plaintext training

	Over plaintext	CARER
Benign	84/86 (97.8%)	84/86 (97.8%)
Malignant	106/114 (93.0%)	106/114 (93.0%)
Overall	190/200 (95.0%)	190/200 (95.0%)

6.3 Computation Cost

In this section, we analyze and test the computation cost of our CARER in terms of two phases, i.e., model training phase and disease risk prediction phase, and make the comparison with the EPDP.

For the sake of simplicity, we only record complex arithmetical operations in this section. Specifically, we set C_{inv}, C_{exp} and C_{mul} to represent the computation cost of a modular inverse, a modular exponentiation and a modular multiplication operation, respectively. Since EPDP uses the bloom filter and symmetric encryption algorithm during the disease query process, for convenience, we use C_H, C_{SE} and C_{SD} to represent the computation cost of a hash function, a symmetric encryption operation, and a symmetric decryption operation, respectively. Besides, we also use m, u, v, and l to represent the number of MCs, all collected attributes, disease classes, and total instances for training disease risk prediction model, respectively.

• Computation cost of our CARER

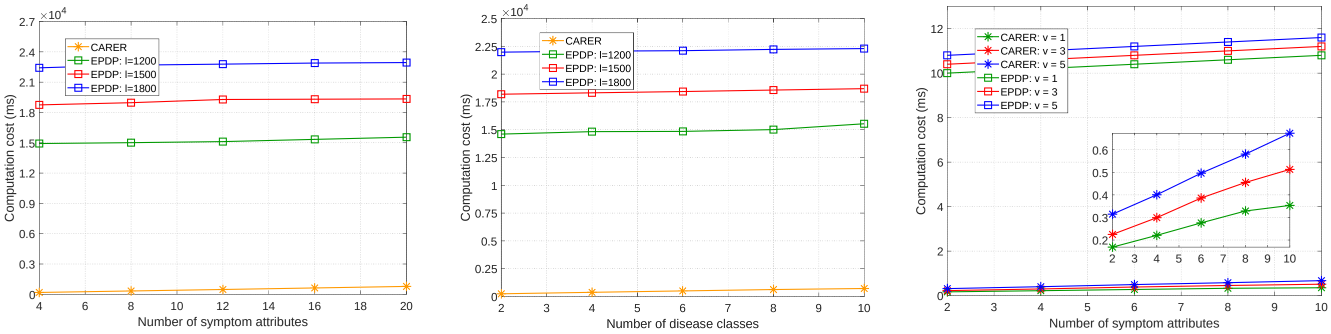
In CARER, during the model training phase, every MC$_i$ encrypts its local training data through executing the modified Paillier cryptosystem, which costs $m \cdot (2uv + 2u + v + 1) \cdot (2C_{exp} + 2C_{mul})$. Then, EP aggregates the ciphertexts from MCs, and executes decryptions for obtaining the global training data, which costs $(m-1) \cdot (2uv + 2u + v + 1) \cdot C_{mul}$ and $(2uv + 2u + v + 1) \cdot (C_{inv} + C_{exp} + 3C_{mul})$, respectively. Therefore, the total computation cost of model training is $(2uv+2u+v+1) \cdot (C_{inv} + (2m+1) \cdot C_{exp} + (3m+2) \cdot C_{mul})$. During the disease risk prediction phase, every U$_i$ generates its query request $C_j, j = 1, 2, \cdots, 2u + 3$, which costs $(4uv + 6) \cdot C_{mul}$. Then, EP computes the query responses $D_t, t = 1, 2, \cdots, v$, in which $4v \cdot (2u + 3) \cdot C_{mul}$ is required. Finally, the U$_i$ costs $2v \cdot C_{mul}$ to obtain the final query results. Therefore, the total computation cost of disease risk prediction is $(8uv + 4u + 14v + 3) \cdot C_{mul}$.

• Computation cost of EPDP

In EPDP, during the model training phase, every data owner costs $3 \cdot (2C_{exp} + C_{mul})$ to encrypt its local training data, thus, it costs $3l \cdot (2C_{exp} + C_{mul})$ totally for l users. Then, for obtaining the global training data, the cloud platform and healthcare provider cost $3 \cdot (l-1) \cdot C_{mul}$ and $3 \cdot (C_{inv} + 2C_{exp} + 3C_{mul})$ to aggregate and decrypt the ciphertexts, respectively. After this, it costs $S \cdot C_H + v \cdot C_{SE}$ for the healthcare provider to generate and encrypt the bloom filter, where S is the number of vectors probably suffering a disease. Therefore, the total computation cost of model training is $6(l+1) \cdot (C_{exp} + C_{mul}) + 3C_{inv} + S \cdot C_H + v \cdot C_{SE}$. During the disease risk prediction phase, each user first generates the encrypted disease risk query request, which costs $C_H + 2C_{exp} + C_{mul}$. Then, the cloud platform computes the

TABLE 3
computation cost of CARER vs EPDP

	Model Training	Disease Risk Prediction
CARER	$(2uv+2u+v+1)\cdot(C_{inv}+2(m+1)\cdot C_{exp}+(3m+2)\cdot C_{mul})$	$(8uv+4u+14v+3)\cdot C_{mul}$
EPDP [17]	$6(l+1)\cdot(C_{exp}+C_{mul})+3C_{inv}+S\cdot C_H+v\cdot C_{SE}$	$3\cdot(C_{exp}+C_{mul})+C_H+T\cdot(C_{SE}+C_{SD})$



(a) Computation cost of model training varying with u and l. ($v = 3$)

(b) Computation cost of model training varying with v and l. ($u = 6$)

(c) Computation cost of data disease risk prediction varying with u and v.

Fig. 4. Computation Cost of CARER vs EPDP.

disease risk query response, in which $C_{exp}+2C_{mul}+T\cdot C_{SE}$ is required, where T is the number of positive diseases. Finally, the user decrypts the query responses to obtain the final query results, which costs $T\cdot C_{SD}$. Therefore, the total computation cost of disease risk prediction is $3\cdot(C_{exp}+C_{mul})+C_H+T\cdot(C_{SE}+C_{SD})$.

- **Comparison**

We present the comparison of computation cost for our CARER and the EPDP in Table 3. From the table, we can see that due to the medical data preprocessing operation, the computation cost of our CARER is only related to the number of medical centers, all collected attributes, and disease classes while it is not affected by the number of instances. Therefore, during the model training phase, the computation cost of CARER is less than that of EPDP when the number of training instances is large. Moreover, during the disease risk prediction phase, the computation cost of EPDP is much higher than our CARER since we simplify the arithmetical operation of naïve Bayesian classification, and only modular multiplication operations are required in CARER.

In Fig. 4, we set the number of MCs $m = 3$, and plot the comparison of computation cost for our CARER and EPDP with the generated synthetic dataset. Specifically, Fig. 4(a) and Fig. 4(b) plot the computation cost comparison during model training phase. Since the computation cost of our CARER is independent of the number of instances and the curves with $l = 1200, 1500$ and 1800 are coincident, we only plot one curve for CARER. Then, in Fig. 4(a) and Fig. 4(b), it can be seen that even if the computation cost of EPDP does not increase with the number of symptoms attributes, our CARER still performs much better than EPDP with different numbers of symptom attributes, disease classes, and instances. Moreover, Fig. 4(c) plots the computation cost comparison during disease risk prediction phase. In detail, we enlarge the computation cost of CARER in Fig. 4(c) since it is too small to be shown clearly. It can be seen that the computation costs of both CARER and EPDP linearly increase with the number of symptom attributes and disease classes, but our CARER is much efficient than EPDP, which demonstrates that our CARER performs much better in practice.

6.4 Communication Overhead

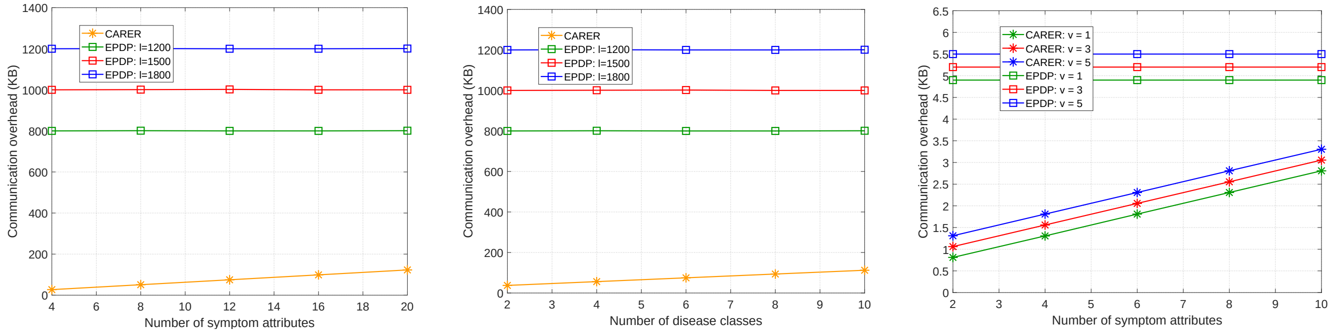
In this section, we analyze and test the communication overhead of our CARER, and then make the comparison with the EPDP.

- **Communication overhead of our CARER**

In CARER, during the phase of model training, each MC$_i$ sends its encrypted local training data $<[\![\mathbf{NX}^{(i)}]\!], [\![\mathbf{NY}^{(i)}]\!], [\![\mathbf{NZ}^{(i)}]\!], [\![\mathbf{ND}^{(i)}]\!], [\![\mathbf{NL}^{(i)}]\!]>$ to EP. Since the security parameter of our modified Paillier cryptosystem is κ, each MC$_i$ spends $2\cdot(2uv+2u+v+1)\cdot\kappa$ bits to outsource the encrypted data to EP. Thus, the communication overhead of model training is $2m\cdot(2uv+2u+v+1)\cdot\kappa$ bits with m MCs. During the phase of disease risk prediction, each U$_i$ first sends her/his encrypted query request $< C_j >, j = 1, 2, \cdots, 2u+3$ to EP, which spends $(2u+3)\cdot k_1$ bits. Then, EP returns the query responses $< D_t, D'_t >, t = 1, 2, \cdots, v$ to U$_i$, which spends $2v\cdot k_1$ bits. Therefore, the communication overhead of disease risk prediction is $(2u+2v+3)\cdot k_1$ bits.

- **Communication overhead of EPDP**

In EPDP, during the phase of model training, it spends $9l\cdot\kappa$ bits for data owners to send their encrypted local training data, and the cloud platform spends 9κ bits to send the aggregated results to the healthcare provider. Moreover, for each disease, the healthcare provider returns a bloom filter to the cloud platform, which spends $v\cdot|BF|$ bits, where $|BF|$ is the length of the bloom filter. Thus, the communication overhead of model training is $(9l+9)\cdot\kappa+v\cdot|BF|$ bits. During the phase of disease risk prediction, the user sends her/his encrypted disease risk query request to the cloud platform, which costs 2κ bits. Then, note that the number



(a) Communication overhead of model training varying with u and l. ($v = 3$)

(b) Communication overhead of model training varying with v and l. ($u = 6$)

(c) Communication overhead of disease risk prediction with u and v.

Fig. 5. Communication Overhead of CARER vs EPDP.

of diseases predicted as positive is T and the bit length of disease name is $|y_t|$, it costs $T \cdot |y_t|$ bits for the cloud platform to send query response to the user. Therefore, the communication overhead of disease risk prediction is $T \cdot |y_t| + 2\kappa$.

TABLE 4
Communication Overhead of CARER vs EPDP

	Model Training	Disease Risk Prediction
CARER	$4m \cdot (2uv + 2u + v + 1) \cdot \kappa$	$(2u + 2v + 3) \cdot k_1$
EPDP [17]	$(9l + 9) \cdot \kappa + v \cdot |BF|$	$T \cdot |y_t| + 2\kappa$

• Comparison

We present the comparison of communication overhead for our CARER and the EPDP in Table 4. Similar to computation cost, we can see that in the phase of model training, the communication overhead of our CARER is independent of the number of total instances while EPDP linearly increases. Therefore, CARER performs much better than EPDP when the number of training instances is large. In the phase of disease risk prediction, the communication of CARER increases with the number of symptom attributes and disease classes, but it is acceptable in practice and still lower than that of EPDP. Besides, our CARER is able to provide a disease risk list of all diseases for users while EPDP is not.

Fig. 5 plots the comparison of communication overhead for our CARER and EPDP with 3 MCs. Specifically, Fig. 5(a) and Fig. 5(b) plot the communication overhead comparison during model training phase. Similar to computation cost, since the communication overhead of our CARER is independent of the number of instances and the curves with $l = 1200, 1500$ and 1800 are coincident, we only plot one curve for CARER. Then, in Fig. 5(a) and Fig. 5(b), it can be seen that communication overhead of our CARER linearly increases with the number of symptom attributes and disease classes while EPDP does not. However, our CARER can maintain a low communication overhead with different number of instances, symptom attributes, and disease classes, while the EPDP spends much more communication overhead than our CARER. Furthermore, Fig. 5(c) plots communication overhead comparison during disease prediction phase. From Fig. 5(c), it can be seen that although the communication overhead of CARER linearly increases with the number of attributes and disease classes, it is still lower that of EPDP, and is acceptable in the real environment.

7 Related Work

In this section, we briefly discuss some related works on privacy-preserving model training and disease risk prediction schemes.

Privacy-preserving model training. Recently, the privacy-preserving data training schemes has been studied widely in e-healthcare system. Li *et al.* [22] introduced a privacy-preserving outsourced classification framework based on fully homomorphic encryption, where the evaluator and crypto service provider can jointly train a naïve bayesian classification model over multi-party outsourced encrypted data. Based on additive homomorphic encryption, Mandal *et al.* [23] designed a method to securely execute gradient descent for data owners and the cloud server, and further achieve the privacy-preserving linear and logistic regression model training. Gascón *et al.* [24] proposed privacy-preserving protocols for training linear regression models, which supports the secure model training over vertically distributed datasets. Shen *et al.* [25] utilized the blockchain techniques to build a secure and reliable data sharing platform among multiple data providers, and constructed a privacy-preserving SVM training scheme based on Paillier encryption system. Wang *et al.* [26] introduced a privacy-preserving collaborative model training scheme on skyline computation, which allows healthcare centers securely train a global diagnosis model with their local medical datasets. Zhou *et al.* [27] proposed a novel secure data processing protocol, which supports both homomorphic addition and multiplication operations over ciphertexts. Based on the proposed protocol, an efficient and privacy-preserving dynamic medical text mining and image feature extraction scheme was proposed. Most above-mentioned schemes only achieve the privacy-preserving data training. Besides, massive interactions are necessary between data providers and cloud servers, which brings heavy communication overhead in practice.

Privacy-preserving disease risk prediction. Nowadays, the privacy-preserving disease risk prediction has also attracted much attention from academia and industry. Ayday *et al.* [28] combined the genomic, clinical and environmental

TABLE 5
Functionality comparison

	Liu *et al* [11]	Yang *et al* [17]	Zhu *et al* [13]	Liu *et al* [18]	Hua *et al* [21]	CARER
Multi-party training	✔	✔	✘	✘	✘	✔
Vertical datasets supporting	✘	✘	✘	✘	✘	✔
Single cloud model	✘	✘	✔	✔	✔	✔
Disease risk list query	✔	✘	✘	✔	✘	✔
Whole process covering	✔	✔	✘	✘	✘	✔
High-efficiency	✘	✔	✔	✔	✔	✔

data, and proposed a privacy-preserving disease risk prediction scheme with homomorphic encryption, which can provide high-accuracy disease risk prediction services for patients. Zhu *et al.* [13], [18], [21], [29] proposed a series of disease prediction schemes based on machine learning techniques, which utilized light-weight privacy-preserving techniques to achieve secure and high-efficiency disease risk prediction. Zhang *et al.* [30] built an efficient index tree for the encrypted genetic data with bloom filters, and introduced a general disease risk prediction framework, which improved the computation and communication performance significantly. Das *et al.* [31] developed a new secret sharing method for protecting the users' private genomic and clinical data during the disease risk prediction, while the authentication of disease risk query results is also ensured. Liu *et al.* [32] proposed a privacy-preserving clinical decision support system through designing a secure single-layer neural network, and extended the proposed scheme to multiple-layer neural network. Ma *et al.* [33] proposed the privacy-preserving random forest classification over encrypted data with secure rational computation protocols for disease prediction, which protected both classifier and user's sensitive data. However, there schemes only deal with the phase of disease risk prediction, which cannot achieve the integrated privacy-preserving from data training to disease risk prediction.

To achieve integrated privacy-preserving, Liu *et al.* [11] proposed a new secure patient-centric clinical decision support system, which preserved the sensitive data in both model training and prediction phases. However, this scheme required complex mathematical operations, which makes it hard to be deployed in practice. Yang *et al.* [17] presented an efficient and secure disease risk prediction scheme based on naïve bayesian classification, which introduced super-increasing sequence to reduce the dimension of datasets and improve the efficiency. Nevertheless, since the coefficients of super-increasing sequences increase exponentially, it is difficult to be used in high-dimensional data encryption. Moreover, there are few of existing schemes working on multi-outsourced vertically distributed datasets in disease risk assessment systems.

Different from existing privacy-preserving disease risk prediction schemes, our proposed CARER achieves integrated privacy-preserving in both model training and disease risk prediction phases. Moreover, CARER is high-efficient in terms of computation cost and communication overhead, and only one cloud is required in our system. In detail, we make a comparison of CARER and existing schemes in TABLE 5. From the table, it can be seen that our CARER is more practical in the real environment.

8 Conclusion

In this paper, we have proposed an efficient and privacy-preserving disease risk assessment scheme over multi-outsourced vertical datasets, called CARER. Based on a modified Paillier cryptosystem and random masking techniques, in CARER, EP can securely train a disease risk prediction model over vertically distributed medical data from multiple medical centers, and provide privacy-preserving disease risk prediction services for users. In this process, all sensitive data of medical centers, e-healthcare provider, and users were well protected. Moreover, the proposed scheme greatly improved the efficiency of privacy-preserving disease risk assessment through data preprocessing and operation transformation. Detailed security analysis showed its security strength and privacy-preserving ability, and extensive experiments were conducted to verify its efficiency.

9 Acknowledgment

This work was supported by National Key Research and Development Program of China (2017YFB0802200), National Natural Science Foundation of China (61972304, 61932015), Natural Science Foundation of Shaanxi Province (2019ZDLGY12-02).

Availability

The implementation of the proposed two schemes and relevant information can be downloaded at `http://xdzhuhui.com/demo/CARER`.

References

[1] G. Manogaran, R. Varatharajan, D. Lopez, P. M. Kumar, R. Sundarasekar, and C. Thota, "A new architecture of internet of things and big data ecosystem for secured smart healthcare monitoring and alerting system," *Future Generation Computer Systems*, vol. 82, pp. 375–387, 2018.

[2] J. Qiu, X. Liang, S. Shetty, and D. Bowden, "Towards secure and smart healthcare in smart cities using blockchain," in *IEEE International Smart Cities Conference*. IEEE, 2018, pp. 1–4.

[3] J. S. Lin, C. V. Evans, E. Johnson, N. Redmond, E. L. Coppola, and N. Smith, "Nontraditional risk factors in cardiovascular disease risk assessment: Updated evidence report and systematic review for the us preventive services task force," *Journal of the American Medical Association*, vol. 320, no. 3, pp. 281–297, 2018.

[4] A. Abbas, M. Ali, M. U. S. Khan, and S. U. Khan, "Personalized healthcare cloud services for disease risk assessment and wellness management using social media," *Pervasive and Mobile Computing*, vol. 28, pp. 81–99, 2016.

[5] S. Perveen, M. Shahbaz, K. Keshavjee, and A. Guergachi, "A systematic machine learning based approach for the diagnosis of non-alcoholic fatty liver disease risk and progression," *Scientific reports*, vol. 8, no. 1, pp. 1–12, 2018.

[6] L. Jena, S. Nayak, and R. Swain, "Chronic disease risk (cdr) prediction in biomedical data using machine learning approach," in *Advances in Intelligent Computing and Communication*, 2020, pp. 232–239.

[7] C. Xu, N. Wang, L. Zhu, K. Sharif, and C. Zhang, "Achieving searchable and privacy-preserving data sharing for cloud-assisted e-healthcare system," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8345–8356, 2019.

[8] W. Tang, J. Ren, K. Deng, and Y. Zhang, "Secure data aggregation of lightweight e-healthcare iot devices with fair incentives," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8714–8726, 2019.

[9] L. Yang, Q. Zheng, and X. Fan, "RSPP: A reliable, searchable and privacy-preserving e-healthcare system for cloud-assisted body area networks," in *2017 IEEE Conference on Computer Communications*. IEEE, 2017, pp. 1–9.

[10] R. Bocu and C. Costache, "A homomorphic encryption-based system for securely managing personal health metrics data," *IBM Journal of Research and Development*, vol. 62, no. 1, pp. 1:1–1:10, 2018.

[11] X. Liu, R. Lu, J. Ma, L. Chen, and B. Qin, "Privacy-preserving patient-centric clinical decision support system on naïve bayesian classification," *IEEE Joutnal of Biomedical and Health Informatics*, vol. 20, no. 2, pp. 655–668, 2016.

[12] K. Y. Yigzaw and J. G. Bellika, "Evaluation of secure multi-party computation for reuse of distributed electronic health data," in *Proceedings of IEEE-EMBS International Conference on Biomedical and Health Informatics*. IEEE, 2014, pp. 219–222.

[13] D. Zhu, H. Zhu, X. Liu, H. Li, F. Wang, H. Li, and D. Feng, "CREDO: efficient and privacy-preserving multi-level medical pre-diagnosis based on ml-*kNN*," *Information Sciences*, vol. 514, pp. 244–262, 2020.

[14] O. Goldreich, S. Micali, and A. Wigderson, "How to play any mental game, or a completeness theorem for protocols with honest majority," in *Providing Sound Foundations for Cryptography: On the Work of Shafi Goldwasser and Silvio Micali*. ACM, 2019, pp. 307–328.

[15] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Advances in Cryptology - EUROCRYPT '99*, J. Stern, Ed., vol. 1592. Springer, 1999, pp. 223–238.

[16] M. Martinez-Arroyo and L. E. Sucar, "Learning an optimal naive bayes classifier," in *18th International Conference on Pattern Recognition (ICPR 2006)*. IEEE Computer Society, 2006, pp. 1236–1239.

[17] X. Yang, R. Lu, J. Shao, X. Tang, and H. Yang, "An efficient and privacy-preserving disease risk prediction scheme for e-healthcare," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3284–3297, 2019.

[18] X. Liu, H. Zhu, R. Lu, and H. Li, "Efficient privacy-preserving online medical primary diagnosis scheme on naive bayesian classification," *Peer-to-Peer Networking and Applications*, vol. 11, no. 2, pp. 334–347, 2018.

[19] R. Lu, H. Zhu, X. Liu, J. K. Liu, and J. Shao, "Toward efficient and privacy-preserving computing in big data era," *IEEE Network*, vol. 28, no. 4, pp. 46–50, 2014.

[20] A. Asuncion and D. Newman, "Uci machine learning repository," 2007. [Online]. Available: https://doi.org/10.1109/MNET.2014.6863131

[21] J. Hua, H. Zhu, F. Wang, X. Liu, R. Lu, H. Li, and Y. Zhang, "CINEMA: efficient and privacy-preserving online medical primary diagnosis with skyline query," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1450–1461, 2019.

[22] P. Li, J. Li, Z. Huang, C. Gao, W. Chen, and K. Chen, "Privacy-preserving outsourced classification in cloud computing," *Cluster Computing*, vol. 21, no. 1, pp. 277–286, 2018.

[23] K. Mandal and G. Gong, "Privfl: Practical privacy-preserving federated regressions on high-dimensional data over mobile networks," in *Proceedings of the 2019 ACM SIGSAC Conference on Cloud Computing Security Workshop*. ACM, 2019, pp. 57–68.

[24] A. Gascón, P. Schoppmann, B. Balle, M. Raykova, J. Doerner, S. Zahur, and D. Evans, "Privacy-preserving distributed linear regression on high-dimensional data," *Proceedings on Privacy Enhancing Technologies*, vol. 2017, no. 4, pp. 345–364, 2017.

[25] M. Shen, X. Tang, L. Zhu, X. Du, and M. Guizani, "Privacy-preserving support vector machine training over blockchain-based encrypted iot data in smart cities," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 7702–7712, 2019.

[26] F. Wang, H. Zhu, X. Liu, R. Lu, J. Hua, H. Li, and H. Li, "Privacy-preserving collaborative model learning scheme for e-healthcare," *IEEE Access*, vol. 7, pp. 166 054–166 065, 2019.

[27] J. Zhou, Z. Cao, X. Dong, and X. Lin, "PPDM: A privacy-preserving protocol for cloud-assisted e-healthcare systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 7, pp. 1332–1344, 2015.

[28] E. Ayday, J. L. Raisaro, P. J. McLaren, J. Fellay, and J. Hubaux, "Privacy-preserving computation of disease risk by using genomic, clinical, and environmental data," in *2013 USENIX Workshop on Health Information Technologies*. USENIX Association, 2013.

[29] H. Zhu, X. Liu, R. Lu, and H. Li, "Efficient and privacy-preserving online medical prediagnosis framework using nonlinear SVM," *IEEE Journal of Biomedical and Health Informatics*, vol. 21, no. 3, pp. 838–850, 2017.

[30] J. Zhang, L. Zhang, M. He, and S. Yiu, "Privacy-preserving disease risk test based on bloom filters," in *19th International Conference on Information and Communications Security*, vol. 10631. Springer, 2017, pp. 472–486.

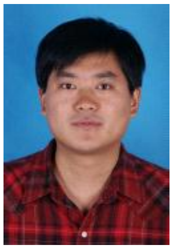
[31] M. Das, N. J. Mozumder, S. Afrose, K. A. Akbar, and T. Hashem, "A novel secret sharing approach for privacy-preserving authenticated disease risk queries in genomic databases," in *2018 IEEE 42nd Annual Computer Software and Applications Conference*. IEEE Computer Society, 2018, pp. 645–654.

[32] X. Liu, R. H. Deng, Y. Yang, N. H. Tran, and S. Zhong, "Hybrid privacy-preserving clinical decision support system in fog-cloud computing," *Future Generation Computer Systems*, vol. 78, pp. 825–837, 2018.

[33] Z. Ma, J. Ma, Y. Miao, and X. Liu, "Privacy-preserving and high-accurate outsourced disease predictor on random forest," *Information Science*, vol. 496, pp. 225–241, 2019.

Fengwei Wang received the B.Sc. degree from Xidian University, Xi'an, China, in 2016. He is current working toward the Master's degree with the School of Cyber Engineering, Xidian University, Xi'an, China.

His research interests include the areas of applied cryptography, cyber security, and privacy.



Hui Zhu (M13) received his B.Sc. degree from Xidian University in 2003, M.Sc. degree from Wuhan University in 2005, and Ph.D. degrees from Xidian University in 2009. In 2013, he was with School of Electrical and Electronics Engineering, Nanyang Technological University as a research fellow.

Since 2016, he has been the professor in the School of Cyber Engineering, Xidian University, China. His research interests include the areas of applied cryptography, data security and privacy.

Rongxing Lu (S'09-M'11-SM'15) is currently an associate professor at the Faculty of Computer Science (FCS), University of New Brunswick (UNB), Canada. Before that, he worked as an assistant professor at the School of Electrical and Electronic Engineering, Nanyang Technological University (NTU), Singapore from April 2013 to August 2016. Rongxing Lu worked as a Postdoctoral Fellow at the University of Waterloo from May 2012 to April 2013. He was awarded the most prestigious "Governor General's Gold Medal", when he received his PhD degree from the Department of Electrical & Computer Engineering, University of Waterloo, Canada, in 2012; and won the 8th IEEE Communications Society (ComSoc) Asia Pacific (AP) Outstanding Young Researcher Award, in 2013. He is presently a senior member of IEEE Communications Society. His research interests include applied cryptography, privacy enhancing technologies, and IoT-Big Data security and privacy. He has published extensively in his areas of expertise, and was the recipient of 8 best (student) paper awards from some reputable journals and conferences. Currently, Dr. Lu currently serves as the Vice-Chair (Conferences) of IEEE ComSoc CIS-TC (Communications and Information Security Technical Committee). Dr. Lu is the Winner of 2016-17 Excellence in Teaching Award, FCS, UNB.

Yandong Zheng received her M.S. degree from the Department of Computer Science, Beihang University, China, in 2017 and she is currently pursuing her Ph.D. degree in the Faculty of Computer Science, University of New Brunswick, Canada.

Her research interest includes cloud computing security, big data privacy and applied privacy.

Hui Li (M10) Received his B.Sc. degree from Fudan University in 1990, M.Sc. and Ph.D. degrees from Xidian University in 1993 and 1998, respectively.

Since 2005, he has been the professor in the school of Telecommunication Engineering, Xidian University, China. His research interests are in the areas of cryptography, wireless network security, information theory and network coding.

Dr. Li served as TPC co-chair of ISPEC 2009 and IAS 2009, general co-chair of E-Forensic 2010, ProvSec 2011 and ISC 2011, honorary chair of NSS 2014, ASI-ACCS 2016.