EAI Endorsed Transactions
on e-Learning
Research Article EAI.EU

Controllable Privacy-Preserving Online Diagnosis with Outsourced SVM over Encrypted Medical Data

Fanxi Wei[1,2], Yuan Ping[2,*], Wenhong Wu[1], Danping Niu[1,2], Yan Cao[2]

[1]School of Information Engineering, North China University of Water Resources and Electric Power, Zhengzhou, China

[2]School of Information Engineering, Xuchang University, Xuchang, China

Abstract

With the widespread application of online diagnosis systems, users can upload their physical characteristics anytime and from anywhere to receive clinical diagnoses. However, for privacy and intellectual property considerations, users' physical characteristics, diagnosis results, and the medical diagnosis model should be protected. To achieve an efficient and secure online diagnosis, secure outsourcing and low burden become research objectives. However, few of the existing privacy-preserving schemes focus on the secure outsourcing of the training process, and few consider the supervision of the hospital for the online diagnosis process. By introducing a four-party architecture with two non-colluding servers, a hospital and users, in this paper, we propose a controllable privacy-preserving online diagnosis scheme (CPPOD) with outsourced SVM over encrypted medical data. Concretely, an integer vector homomorphic encryption is employed to protect medical data and user requests. In the encrypted domain, a series of collaborative protocols including data collection, sequence minimum optimization solver, SVM model building, and online diagnosis are constructed and take place between different participants, while no significant increase in computation on either the hospital or user side. CPPOD enables the hospital to delegate online diagnosis services to a cloud server while ensuring that its regulatory capabilities cannot be bypassed unauthorized. Security analysis and performance evaluation suggest that CPPOD performs well regarding security and efficiency.

Keywords: Support vector machine, Secure outsourcing, Vector homomorphic encryption, Privacy-preserving online diagnosis

Received on 17 November 2023; accepted on 02 December 2023; published on 07 December 2023

doi:10.4108/eetel.4412

1. INTRODUCTION

Motivated by rapid data accumulation and the requirement of convenient medical services, online medical diagnosis has attracted much attention[1–3]. By introducing different machine learning methods, various online diagnosis schemes have been proposed, e.g.[4–8]. As a readily available diagnostic service, it effectively reduces the cost of the medical system and has contributed significantly to the advancement of the medical industry [9, 10]. As one of the most potent classification algorithms, support vector machine (SVM) exhibits high efficiency in dealing with sparse and high-dimensional data. Moreover, they demonstrate exceptional accuracy and effectiveness in classification tasks, making them widely used in the field of medical diagnosis [7, 11–13]. To establish a comprehensive online diagnosis system, the hospital usually trains an SVM-based diagnosis model and then outsources the trained model to the cloud. Users can access the online diagnosis service by transmitting physical characteristics to the cloud and receiving the diagnosis suggestion.

With the cloud's assistance, not only the burden of the hospital can be alleviated, but it also enhances user experience. However, despite many benefits with the cloud, outsourcing the model to a semi-trusted cloud also raises serious privacy concerns [14]. For patients, sensitive information may be included in their physical characteristics that should not be leaked

*Corresponding author. Email: pingyuan@xcu.edu.cn

to the cloud. For the hospital, the clinical decision model is also a valuable intellectual property that should be protected. Any information leaked can cause irreversible consequences.

For online medical diagnostics, therefore, the main challenges are how to achieve secure outsourcing, privacy protection, and low burden. That is, we have to ensure that the users' private information and diagnostic decisions are not known to the cloud, and the diagnostic model is also kept secret from the cloud server [15]. Meanwhile, we should also reduce the computational complexity and communication complexity of the hospital and the user as much as possible. Towards these objectives, most of the existing schemes introduce homomorphic encryption (HE) [16, 17] and secure multi-party computation (MPC) [18, 19]. Methods based on HE can perform the necessary calculations in the encrypted domain that can support privacy protection. At present, partial homomorphic encryption (PHE) built on algorithms such as RSA, ElGamal, and Paillier are preferred[20]. By contrast, MPC allows multiple parties to jointly compute a function without revealing their inputs and ensures that each party only obtains its calculation results and cannot infer the input and output data of any other party through the interactive data in the computation process. However, few consider diagnosis model training and diagnostic decision with more than a third-party cloud's assistance. Moreover, most of them hand over the trained model directly to a server for online diagnosis without considering the supervision of the hospital for the online diagnosis services.

Considering the above issues, in this paper, we propose a controllable privacy-preserving online diagnosis scheme (CPPOD) with outsourced SVM over encrypted medical data. Firstly, inspired by [21, 22], we consider a four-party architecture with two non-colluding cloud servers, i.e., users, a hospital, a computing server, and a key server. Users provides pathological data, the hospital protects the data, and cooperates with the computing server to complete the model construction in encrypted domain (ED). Then, the computing server utilizes the model to provide online diagnostic services. The key server is responsible for the key management and identity authentication. Secondly, to train the diagnosis model, we have designed a complete protocol based on the HE of integer vectors [23, 24], which conducts collaborative training of the hospital and the computing server. Finally, with a specifically designed decision outsourcing protocol, the hospital can supervise the online diagnostic service without being bypassed. The main contributions can be summarized as follows.

(i) A secure outsourcing protocol for the diagnosis model training phase is designed. By introducing the HE on integer vectors, the outsourcing protocol supports a collaborative model training between the hospital and the cloud server in ED. So, pricey computations towards the kernel function and model training can be securely transferred to the cloud server equipped with sufficient computing resources from the hospital.

(ii) We present an improved decision outsourcing protocol that ensures the hospital's regulation over the online diagnostic service. Through an indispensable phase design in the online diagnosis process, the lightweight participation of the hospital can not be bypassed without authorization in the four-party collaboration scenario. It not only ensures data privacy but also protects the rights and interests of the hospital as the model owner from the perspective of intellectual property.

(iii) Extensive experiments on real-world datasets are conducted to evaluate the proposed CPPOD, e.g., Breast Cancer Wisconsin (BCW), Maternal Health Risk Data (Maternal), and HCV data. Both performances in plain domain (PD) and ED are considered. Experimental results demonstrate that CPPOD can significantly avoid pricey computation on both the hospital and user, and prevent model abuse by the cloud server while guaranteeing data privacy and comparable accuracies with working in PD.

Unlike most of the traditional SVM-based privacy-preserving online diagnosis schemes, CPPOD supports secure outsourcing of both the model training and online diagnosis phases, and it guarantees the hospital's regulation over online diagnostic service performed on the cloud server. In addition, CPPOD is secure and efficient in terms of time and communication. In Table 1, we compare CPPOD with several representative schemes from the model architecture and the implemented functions.

The remainder of this paper is organized as follows. Section 2 introduces the related works. Section 3 provides essential definitions and relevant background. Then, Section 4 gives the system model whose critical protocols and implementations are detailed in Section 5. In Section 6 and 7, we analyze CPPOD's security and performance, respectively. Experimental results are presented and discussed in Section 8. Finally, the conclusion is drawn in Section 9.

2. RELATED WORK

With the construction of smart hospitals, many hospitals apply digital and healthcare big data to provide data services, such as intelligent pre-consultation, prescription dispensing, and clinical assistant decision-making. As one of the representative services, online

Table 1. Model Architecture and Phase Implementation Comparison

	Rahulamathavan et al[25]	Zhang et al.[26]	Chen et al.[12]	Chen et al.[27]	Wang et al.[28]	Xie et al.[29]	CPPOD
model architecture	2-party	3-party	3-party	4-party	6-party	4-party	4-party
Secure outsourcing of the model training	–	✗	–	✓	✓	✗	✓
Secure outsourcing of the model testing	✓	✓	✓	–	✓	✓	✓
Regulation of online services	✗	✗	✗	–	✗	✗	✓

diagnosis offers medical suggestions for users by employing machine learning models on the cloud that greatly facilitate users and hospitals. However, privacy concerns can not be avoided even though it brings convenience. Many different privacy-preserving machine learning schemes have been proposed [30–32]. Specifically, based on naive Bayesian classification, Wang et al. [30] proposed an efficient privacy-preserving disease risk assessment scheme. In the scheme, an improved Paillier cryptosystem is designed for secure model training, and the random mask technology is used to provide users with privacy-preserving disease risk prediction services. By introducing improved KNN computation and elementary matrix permutation, Zhang et al. [31] constructed a privacy-preserving decision tree evaluation scheme for e-health systems. Liu et al. [32], based on Paillier homomorphic cryptosystem and secret sharing, designed a series of building blocks that perform secure computation under the two-cloud model for privacy-preserving KNN classification. In the literature[20], SVM is widely preferred for clinical decisions in online diagnosis systems for its solid mathematical background and high efficiency in dealing with high-dimensional and complex data. These solutions can be divided into HE-based schemes and MPC-based schemes.

HE is a particular encryption scheme that allows any third party to operate on encrypted data without decrypting it beforehand. For instance, by employing additively homomorphic encryption, Rahulamathavan et al. [25] proposed a privacy-preserving decision support system for SVM with a Gaussian kernel. Although it protects diagnosis data and results, there are still some robustness and security issues, such as repetitive attacks[33]. Later, based on the Okamoto-Uchiyama (OU) cryptosystem, Zhang et al. [26] constructed a three-party model architecture of privacy-preserving multi-class SVM over encrypted data for medical diagnosis assistance. It not only protects the medical dataset but also ensures that the cloud server does not know the diagnosis results. However, the user has to request the model parameters from the hospital, compute the decision function by himself in ED, and upload the encrypted diagnosis results to the cloud server for the final decision. Since users do many works, the user side has to afford a relatively large computational cost. Recently, Chen et al. [12] constructed a privacy-preserving medical diagnosis scheme using distributed two trapdoors public key cryptosystem (DT-PKC) and Boneh-Goh-Nissim (BGN) cryptosystem. Since the main objective is to give a secure implementation of the diagnosis process, it suits SVMs with different kernel functions. However, the scheme cannot endow essential regulatory capability to the hospital, and the efficiency improvement on either the user or hospital side is insignificant.

MPC is a general cryptographic primitive that provides privacy protection for cooperative computation. It deals securely with the problem of several parties with private data performing collaborative analysis in distributed computing scenarios. It is widely used for SVM-based online diagnosis. For example, Chen et al. [27] privately train SVM on outsourced genomic data via MPC. Using the semi-honest adversary model and oblivious transfer, they train non-linear SVM on the combined data of multiple data sources without sacrificing personal privacy. However, as the model owner, the hospital has to undertake a relatively significant computation burden. Towards a better solver of data privacy in the medical internet of things, Wang et al. [28] designed eight secure computing protocols to make the cloud server efficiently execute basic integer and floating-point computations. Thus, gradient descent (GD) based SVM training can be achieved in a six-party scenario. Due to complex roles with different abilities with respect to data, intermediate results, and models, the correct online diagnosis process may be vulnerable to interference from malicious adversaries, and the service provider may be bypassed by the cloud storage server. Another insightful and user-friendly online diagnosis model was proposed by Xie et al. [29] based on two non-colluding cloud servers with high security and high efficiency in computation and communication. In the scheme, a class of secure two-party protocols using HE was proposed to support the diagnosis decision outsourcing based on a multi-class SVM.

Among the above SVM-based schemes, few of them focus on secure outsourcing both diagnosis model training and decision processes. Even though data privacy can be guaranteed, for most schemes, the user or hospital side has to afford a relatively large amount of computation. Due to the architecture setting, the cloud server frequently acts as a service

agent not a computation agent, throughout the online medical diagnosis process. Once we directly migrate the trained model to the cloud server to remit the hospital's computational burden, the hospital can hardly control or regulate the diagnosis process. As a response to these issues, in this study, our CPPOD prefers a four-party online diagnosis system with two non-colluding cloud servers, one of which is responsible for complex computing tasks and online diagnostic services (computing server), and the other is responsible for key management (key server). Meanwhile, CPPOD adopts an integer vector-supported HE to match the linear SVM, performs the privacy-preserving model training between the hospital and the computing server, and makes diagnosis decisions between the user and the computing server under the lightweight control of the hospital. Although the major computations are conducted on the computing server, the intellectual property of the medical diagnostic model owned by the hospital can be guaranteed because unauthorized bypassing of the hospital will lead to unusable diagnostic results.

3. PRELIMINARY

In this section, we briefly introduce vector homomorphic encryption (VHE) and SVC to be employed in the proposed CPPOD. All the notations are summarized in Table 2.

3.1. Vector Homomorphic Encryption

Designed by [23], VHE operates directly on integer vectors that supports three fundamental operations: addition, linear transformation, and weighted inner products.

Let $\mathbf{x} \in \mathbb{Z}_l^m$ be the integer vector of length m and alphabet size l. Let $\mathbf{c} \in \mathbb{Z}_q^n$ be the ciphertext of $\mathbf{x}$ with length $n > m$ and alphabet size $q \gg l$. In general, q is super-polynomial in the ciphertext length n. The secret key is a matrix $\mathbf{S} \in \mathbb{Z}_q^{m\times n}$, and the process of encrypting $\mathbf{x}$ is to find a ciphertext $\mathbf{c}$ such that $\mathbf{Sc}$ satisfies

$$\mathbf{Sc} = q\mathbf{k} + r\mathbf{x} + \mathbf{e}, \tag{1}$$

for some integer vector $\mathbf{k}$ and noise vector $\mathbf{e}$. Here, r is an integer parameter that satisfies $r > 2|\mathbf{e}|$. $|\cdot|$ returns the maximum absolute value of the elements in vector "$\cdot$". Given the secret key $\mathbf{S}$, decrypting $\mathbf{c}$ can be formulated by

$$\mathbf{x} = \lceil \frac{\mathbf{Sc}}{r} \rfloor_q, \tag{2}$$

where $\lceil a \rfloor_q$ means the nearest integer to a with modulus q. If $|\mathbf{e}|$ is smaller than $\frac{r}{2}$, the decryption is successful. So, we consider that both $|\mathbf{S}|$ and $|\mathbf{e}|$ are much smaller than r.

Key-switching in VHE allows us to change the original secret key into another one we specify (which satisfies certain conditions). Of course, we also need to change the ciphertext accordingly so that the switched ciphertext can be decrypted with the new secret key to obtain the same plaintext as the original. Following [23], we formulate the key-switching method in two steps. (1) Step one: The secret key $\mathbf{S}$ is switched to an intermediate secret key $\mathbf{S}^*$ and the corresponding ciphertext is $\mathbf{c}^*$. After key-switching, we have $\mathbf{S}^*\mathbf{c}^* = \mathbf{Sc}$. (2) Step two: Towards switching the intermediate secret key $\mathbf{S}^* \in \mathbb{Z}^{m\times n\ell}$ to a desired secret key $\mathbf{S}' \in \mathbb{Z}_q^{m\times n'}$, we construct an integer matrix $\mathbf{M} \in \mathbb{Z}^{n'\times n\ell}$ satisfying

$$\mathbf{S}'\mathbf{M} = \mathbf{S}^* + \mathbf{E} \mod q \tag{3}$$

where $\mathbf{E}$ is a noise matrix with a small magnitude. If $\mathbf{S} = [\mathbf{I}, \mathbf{T}]$ with an identity matrix $\mathbf{I}$, $\mathbf{M}$ can be constructed by

$$\mathbf{M} = \begin{pmatrix} -\mathbf{TA} + \mathbf{S}^* + \mathbf{E} \\ \mathbf{A} \end{pmatrix} \mod q, \tag{4}$$

where $\mathbf{A} \in \mathbb{Z}_q^{(n'-m)\times n\ell}$ is a random matrix. Corresponding to the new secret key $\mathbf{S}'$, we have a new ciphertext

$$\mathbf{c}' = \mathbf{M}\mathbf{c}^* \mod q, \tag{5}$$

satisfying

$$\mathbf{S}'\mathbf{c}' = q\mathbf{k}' + r\mathbf{x} + \mathbf{e}'. \tag{6}$$

Let $\mathbf{c}_1$, $\mathbf{c}_2$ be the two ciphertexts of integer vectors $\mathbf{x}_1$, $\mathbf{x}_2$ separately encrypted by secret keys $\mathbf{S}_1$, $\mathbf{S}_2$. They satisfy

$$\mathbf{S}_i\mathbf{c}_i = q\mathbf{k}_i + r\mathbf{x}_i + \mathbf{e}_i, \tag{7}$$

with $|\mathbf{S}_i|$, $|\mathbf{k}_i|$ and $|\mathbf{e}_i|$ much smaller than q. Three fundamental operations supported by VHE are as follows.

- **Addition Operation**: If $\mathbf{c}_1$ and $\mathbf{c}_2$ have the same secret key, i.e., $\mathbf{S}_1 = \mathbf{S}_2 = \mathbf{S}$, then

$$\mathbf{c}_1 + \mathbf{c}_2 = [\![\mathbf{x}_1 + \mathbf{x}_2]\!] \tag{8}$$

- **Linear Transformation**: The ciphertext $\mathbf{c}_1'$ of the linear transformation $\mathbf{Gx}_1$ can be formulated by

$$\mathbf{c}_1' = \mathbf{c}_1 = [\![\mathbf{G}\mathbf{x}_1]\!], \tag{9}$$

with an switched secret key $\mathbf{S}_1' = \mathbf{G}\mathbf{S}_1'$.

- **Weighted Inner Products**: By introducing a key-switching matrix $\mathbf{M}$, the ciphertext of a weighted inner products $\mathbf{x}_1^T\mathbf{H}\mathbf{x}_2$ can be calculated by

$$\mathbf{M}\lceil \frac{\mathrm{vec}\left(\mathbf{c}_1\mathbf{c}_2^{\mathrm{T}}\right)}{r} \rfloor_q = [\![\mathbf{x}_1^T\mathbf{H}\mathbf{x}_2]\!], \tag{10}$$

where $\mathbf{H}$ is the weight matrix.

3.2. Support Vector Machine

Let D be a dataset with N data samples $\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_N, y_N)\}$, where $\mathbf{x}_i \in \mathbb{R}^m$, $y_i \in \mathcal{Y} = \{+1, -1\}$, $i = 1, 2, \ldots, N$. The fundamental concept of a linear SVM for classification is to find the optimal separating hyperplane $\mathbf{w} \cdot \mathbf{x} + b = 0$ in the feature space, which maximizes the interval between positive and negative training samples by solving the following dual problem.

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j \mathbf{x}_i^{\mathrm{T}}\mathbf{x}_j - \sum_{i=1}^{N}\alpha_i \\ \text{s.t.} \quad & \sum_{i=1}^{N}\alpha_i y_i = 0 \\ & 0 \leqslant \alpha_i \leqslant \mathrm{C}, \quad i = 1, 2, ..., N \end{aligned} \tag{11}$$

where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_N]^{\mathrm{T}}$ is the Lagrange multiplier vector, C is the penalty parameter. Thus, we have

$$\mathbf{w} = \sum_{i=1}^{N}\alpha_i y_i \mathbf{x}_i, \tag{12}$$

and

$$b = y_j - \sum_{i=1}^{N} y_i \alpha_i < \mathbf{x}_i \cdot \mathbf{x}_j > . \tag{13}$$

Based on the prediction function $f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$, the decision function $g(\mathbf{x})$ is

$$\begin{aligned} g(\mathbf{x}) &= \mathrm{sign}(\mathbf{w} \cdot \mathbf{x} + b) \\ &= \mathrm{sign}\left(\sum_{i=1}^{N}\alpha_i y_i < \mathbf{x}_i \cdot \mathbf{x} > + b\right). \end{aligned} \tag{14}$$

If D is not linearly separable, we can replace the inner product $< \mathbf{x}_i \cdot \mathbf{x} >$ by a nonlinear kernel function such as radial basis function $\exp(-q\|\mathbf{x}_i - \mathbf{x}\|^2)$ to construct a nonlinear classifier. q is the kernel width.

4. SYSTEM MODEL

In this paper, we consider a four-party framework for outsourcing the privacy-preserving online diagnosis service to the cloud. As shown in Figure.1, four entities include two non-colluding cloud servers $\mathrm{CS_A}$ and $\mathrm{CS_B}$, the hospital H, and users Us.

(i) **Two cloud servers ($\mathrm{CS_A}$ and $\mathrm{CS_B}$):** $\mathrm{CS_A}$ and $\mathrm{CS_B}$ are two non-colluding cloud servers. $\mathrm{CS_A}$ is equipped with sufficient computation, communication, and storage resources for model training in cooperation with H. Meanwhile, given the trained model, $\mathrm{CS_A}$ provides online diagnosis services for Us. $\mathrm{CS_B}$ is mainly responsible for key management and identity authentication.

Table 2. Notations and Description

Notations	Descriptions
$\mathbf{x}$	Data sample in the form of integer vector with length m
l	The alphabet size of $\mathbf{x}$
$\mathbf{c}$	The ciphertext of $\mathbf{x}$ with length n
q	The alphabet size of $\mathbf{c}$, and $q \gg l$
$\mathbf{S}$	The secret key in $\mathbb{Z}_q^{m\times n}$
r	An integer parameter satisfies $r > 2\|\mathbf{e}\|$
$\mathbf{e}$	The noise vector
$\mathbf{S}^*$	Intermediate secret key of $m \times n\ell$ matrix
$\mathbf{c}^*$	Intermediate ciphertext with length $n\ell$
ℓ	An integer satisfies $2^\ell > \|c_k\|$ wherec_k is the member of $\mathbf{c}$ with the largest absolute value
$\mathbf{S}'$	The desired secret key of a $m \times n'$ matrix
$\mathbf{c}'$	The ciphertext encrypted by $\mathbf{S}'$ with length n'
$\mathbf{M}$	The $n' \times n\ell$ key-switch matrix
PK_{H}, SK_{H}	The public key and private key of H
PK_{U}, SK_{U}	The public key and private key of Us
PK_{A}, SK_{A}	The public key and private key of $\mathrm{CS_A}$
K_{S}, K'_{S}	The session key
$[\![\cdot]\!]$	The encrypted form of data
ID	The user ID who uploads $\mathbf{x}$
Rd	A random response number generated for user
$\|\|$	Message connection symbol
D	The original medical dataset
D'	The processed medical dataset
$\mathbf{Q}$	A random orthogonal matrix
$\mathbf{K}$	The kernel matrix
$\boldsymbol{\alpha}$	The optimal solution to the dual problem
$\mathbf{w}$, b	The model parameters
R	The prediction result given by $\mathrm{CS_A}$
d	The final clinical diagnosis
N	The number of m-dimensional data
N_{Us}	The number of data samples uploaded by Us
N_{U_i}	The number of request data samples submitted by Us
N_{class}	The number of classes

(ii) **Hospital (H):** As a user data agent, H collects medical data from a large number of Us to form the medical dataset D, and conduct the model training together with $\mathrm{CS_A}$. Data privacy is guaranteed throughout the two phases. In addition, H authorizes $\mathrm{CS_A}$ to provide online

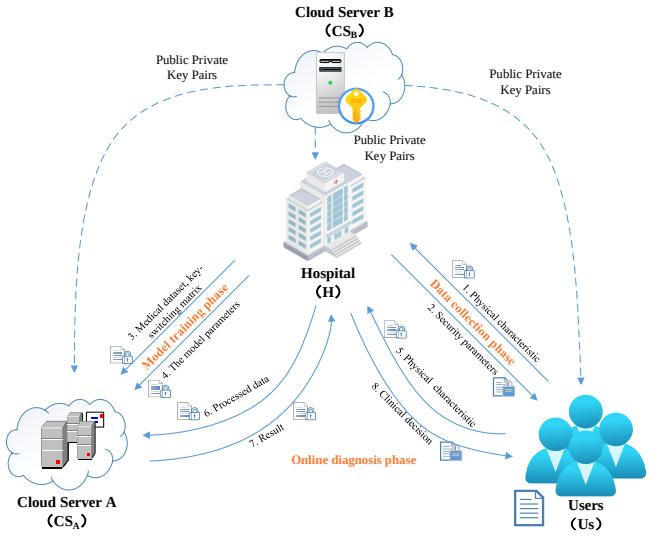


Figure 1. System Model

diagnosis services and must be able to get the business volume externally provided by CS_A.

(iii) **Users (**Us**):** Us are entities who have some physical discomfort and want to query a clinical diagnosis. They submit the physical characteristics and receive the diagnosis results.

Utilizing a four-party architecture, the proposed scheme encompasses three distinct operational phases: data collection, model training, and online diagnosis. During the data collection phase, Us transmit their physical characteristics to H, which reciprocates the security parameters to Us. The model training phase is collaboratively undertaken by H and CS_A. H dispatches the medical dataset to CS_A and the collaborative model training ensues the complex computation tasks are done by CS_A without decrypting any medical data. Finally, H will send the model parameters to CS_A which is authorized to provide online diagnosis services. In the online diagnosis phase, once received the request data from Us, H processes these data and sends them to CS_A. Thus, CS_A makes predictions and returns the results to H who will determine the final clinical diagnosis results and send them to Us. As depicted in Figure.1 with locks, all the data transferred among the four parties are in ED.

In the online diagnosis system, H and Us are honest entities. CS_A and CS_B are non-colluding, but CS_A acts in an honest-but-curious way. It means that CS_A follows the designed protocols honestly but tries to learn more information during the training of the model and the execution of the online diagnosis service. So, we should preserve the privacy of the diagnosis result and medical data against CS_A, and the diagnosis model is unknown by CS_A.

5. THE PROPOSED CPPOD

As depicted by Figure.1, the four-party scheme mainly consists of three work phases, i.e., data collection, model training, and online diagnosis. Before entering these three phases, essential authentications for CS_A, H, and Us should be passed that are suggested yet omitted for out of this study's scope. Thus, these three phases will be detailed below.

5.1. Data collection

In the data collection phase, communication mainly takes between H and Us. H collects the physical characteristics from Us and stores them in the medical dataset D. Due to privacy concerns, $\mathbf{x}$ is encrypted before being sent to H. As the data owner, Us encrypt $\mathbf{x}$ with a session key K_S generated by himself/herself to get $[\![\mathbf{x}]\!]$, then K_S is encrypted with the public key of H (PK_H) to get $[\![K_S]\!]$. After receiving $[\![\mathbf{x}]\!]$ and $[\![K_S]\!]$, H firstly d ecrypts $[[K_S]\!]$ u sing t he p rivate k ey S K_H to get K_S, by which $[\![\mathbf{x}]\!]$ can be decrypted for analysis or correction. Meanwhile, H can give the user a response for confirmation. In the considered scenario, we suggest that the hospital should have full control over the pathological data. Otherwise, the diagnosis model can not be learned with correctly labeled data for online services. The whole procedure of the data collection phase is presented by Algorithm 1. Line 8 is an optional response from H.

Algorithm 1 Data collection

Input: data sample $\mathbf{x}$ of physical characteristic
Output: medical dataset D
Us:
1: Random generation of K_S
2: $[\![\mathbf{x}]\!] = \mathrm{Enc}(K_S, \mathbf{x})$
3: $[\![K_S]\!] = \mathrm{Enc}(PK_H, K_S)$
4: Us $\rightarrow$ H: $[\![\mathbf{x}]\!]$ and $[\![K_S]\!]$
H:
5: $K_S = \mathrm{Dec}(SK_H, [\![K_S]\!])$
6: $\mathbf{x} = \mathrm{Dec}(K_S, [\![\mathbf{x}]\!])$
7: $D \leftarrow D \cup \mathbf{x}$
8: H $\rightarrow$ Us : $[\![Rd||ID]\!] = \mathrm{Enc}(PK_U, Rd||ID)$

5.2. Model training

In the model training phase, the communication is mainly between H and CS_A. By introducing VHE, CPPOD prefers the classical sequence minimum optimization (SMO) algorithm to solve the dual problem (11) in the training process. In addition, we adopt an essential data preprocessing before outsourcing data to CS_A. Next, we will show how this stage is implemented in detail.

SMO in PD. As a heuristic algorithm, SMO is to decompose the original quadratic programming problem into a quadratic programming subproblem containing only two variables and constantly solve the subproblem until all the variables satisfy the KKT condition[34]. It mainly consists of two parts: an analytical method for solving two-variable quadratic programming and a heuristic method for selecting variables.

Without loss of generality, we use K_{ij} to denote $K(\mathbf{x}_i, \mathbf{x}_j)(i, j \in [1, N])$. For linear SVC classifier, $K(\mathbf{x}_i, \mathbf{x}_j)$ is the inner product $< \mathbf{x}_i \cdot \mathbf{x}_j >$. To solve the dual problem (11) , we choose α_1 and α_2 as the optimization objects satisfying $\sum_{i=1}^{N} y_i\alpha_i = 0$, i.e., $\alpha_1 y_1 + \alpha_2 y_2 = -\sum_{i=3}^{N} y_i\alpha_i = \xi$. Thus, we get $\alpha_1 = (\xi - \alpha_2 y_2)y_1$. Suppose $v_j = \sum_{i=3}^{N} y_i\alpha_i K_{ij}$ and $\sum_{i=3}^{N} \alpha_i = Constant$, the dual problem can be reformulated by

$$\begin{aligned}\min_{\alpha_2} W(\alpha_2) =& \frac{1}{2}K_{11}(\xi - \alpha_2 y_2)^2 + \frac{1}{2}K_{22}\alpha_2^2 \\ &+ y_2 K_{12}(\xi - \alpha_2 y_2)\alpha_2 + v_1(\xi - \alpha_2 y_2) \\ &+ y_2 v_2 \alpha_2 - y_1(\xi - \alpha_2 y_2) \\ &- \alpha_2 - Constant\end{aligned} \tag{15}$$

By solving (15), we can get

$$\begin{aligned}&(K_{11} + K_{22} - 2K_{12})\alpha_2 \\ &= y_2(y_2 - y_1 + \xi K_{11} - \xi K_{12} + v_1 - v_2)\end{aligned} \tag{16}$$

Consider the prediction model $f(\mathbf{x})$, we have the predicted deviation

$$E_i = \left(\sum_{j=1}^{N} \alpha_j y_j K_{ji} + b\right) - y_i, \quad j = 1, 2. \tag{17}$$

Then, Eq. (16) can be reformulated by

$$\begin{aligned}&(K_{11} + K_{22} - 2K_{12})\alpha_2^{\text{new}} \\ &= y_2(E_1 - E_2) + (K_{11} + K_{22} - 2K_{12})\alpha_2^{\text{old}}.\end{aligned} \tag{18}$$

Let $\eta = K_{11} + K_{22} - 2K_{12}$, we can get

$$\alpha_2^{\text{new}} = \alpha_2^{\text{old}} + \frac{y_2(E_1 - E_2)}{\eta} \tag{19}$$

Assume that the upper and lower boundaries of α_2 are H and L, respectively. According to $\alpha_1 y_1 + \alpha_2 y_2 = \xi$ and $0 \le \alpha_1, \alpha_2 \le C$, we have $L = \max(0, \alpha_2 - \alpha_1)$ and $H = \min(C, C - \alpha_1 + \alpha_2)$ if $y_1 \ne y_2$, or $L = \max(0, \alpha_2 + \alpha_1 - C)$ and $H = \min(C, \alpha_1 + \alpha_2)$ if $y_1 = y_2$. Therefore α_2^{new} can be obtained by

$$\alpha_2^{\text{new}} = \begin{cases} H, & \alpha_2^{\text{new}} > H \\ \alpha_2^{\text{new}}, & L \le \alpha_2^{\text{new}} \le H \\ L, & \alpha_2^{\text{new}} < L \end{cases} \tag{20}$$

Due to $\alpha_1^{\text{new}} y_1 + \alpha_2^{\text{new}} y_2 = \alpha_1^{\text{old}} y_1 + \alpha_2^{\text{old}} y_2$, we can get $\alpha_1^{\text{new}} = \alpha_1^{\text{old}} + y_1 y_2(\alpha_2^{\text{old}} - \alpha_2^{\text{new}})$. Since b is directly related to $f(\mathbf{x})$, along with α changes, b should be updated as follows

$$b^{\text{new}} = \begin{cases} b_1^{\text{new}} & 0 < \alpha_1^{\text{new}} < C \\ b_2^{\text{new}} & 0 < \alpha_2^{\text{new}} < C \\ \frac{b_1^{\text{new}} + b_2^{\text{new}}}{2} & \text{otherwise,} \end{cases} \tag{21}$$

where

$$\begin{aligned}b_1^{\text{new}} = b^{\text{old}} &- E_1 - y_1 K_{11}(\alpha_1^{\text{new}} - \alpha_1^{\text{old}}) \\ &- y_2 K_{21}(\alpha_2^{\text{new}} - \alpha_2^{\text{old}})\end{aligned} \tag{22}$$

and

$$\begin{aligned}b_2^{\text{new}} = b^{\text{old}} &- E_2 - y_1 K_{12}(\alpha_1^{\text{new}} - \alpha_1^{\text{old}}) \\ &- y_2 K_{22}(\alpha_2^{\text{new}} - \alpha_2^{\text{old}}).\end{aligned} \tag{23}$$

According to the formula derivation, the whole SMO algorithm for the dual problem (11) in PD can be completed after several iterations.

SMO in ED. Since VHE deals with integer vectors, data samples of D and model parameters outsourced to CS_A should be encrypted in vectors. Therefore, to conduct SMO solver in ED, $[\![E_i]\!]$, $[\![\boldsymbol{\alpha}]\!]$ and $[\![b]\!]$ should also be correctly obtained in ED. In this section, we formulate them and describe the corresponding SMO algorithm in ED (SMO-ED).

Following (17) with VHE, $[\![E_i]\!]$ can be formulated by

$$[\![E_i]\!] = \sum_{j=1}^{N} [\![\alpha_j y_j K_{ji}]\!] + [\![b]\!] - [\![y_i]\!], \tag{24}$$

where

$$\begin{cases} [\![\alpha_j y_j K_{ji}]\!] = \frac{1}{m}\lceil \frac{\text{vec}([\![\alpha_j y_j]\!][\![K_{ji}]\!]^{\mathrm{T}})}{r} \rfloor_q, \\ [\![\alpha_j y_j]\!] = \frac{1}{m}\lceil \frac{\text{vec}([\![\alpha_j]\!][\![y_j]\!]^{\mathrm{T}})}{r} \rfloor_q. \end{cases} \tag{25}$$

Meanwhile, given $[\![\eta]\!] = [\![K_{ii}]\!] + [\![K_{jj}]\!] - 2[\![K_{ij}]\!]$, $[\![\alpha_j]\!]$ can be obtained by

$$[\![\alpha_j]\!] = [\![\alpha_j^{old}]\!] + [\![\frac{y_j(E_i - E_j)}{\eta}]\!], \tag{26}$$

with

$$\begin{cases} [\![y_j(E_i - E_j)]\!] = \frac{1}{m}\lceil \frac{\text{vec}([\![y_j]\!]([\![E_i]\!] - [\![E_j]\!])^{\mathrm{T}})}{r} \rfloor_q, \\ [\![\frac{y_j(E_i - E_j)}{\eta}]\!] = \frac{1}{m}\lceil \frac{\text{vec}([\![y_j(E_i - E_j)]\!][\![\frac{1}{\eta}]\!]^{\mathrm{T}})}{r} \rfloor_q. \end{cases} \tag{27}$$

Given $[\![\alpha_j]\!]$, $[\![\alpha_i]\!]$ thus can be updated through

$$[\![\alpha_i]\!] = [\![\alpha_i^{\text{old}}]\!] + [\![y_i y_j(\alpha_j^{\text{old}} - \alpha_j)]\!], \tag{28}$$

with

$$\begin{cases} [\![y_i y_j]\!] = \frac{1}{m}\lceil \frac{\text{vec}([\![y_i]\!][\![y_j]\!]^{\mathrm{T}})}{r} \rfloor_q, \\ [\![y_i y_j(\alpha_j^{\text{old}} - \alpha_j)]\!] = \frac{1}{m}\lceil \frac{\text{vec}([\![y_i y_j]\!]([\![\alpha_j^{\text{old}}]\!] - [\![\alpha_j]\!])^{\mathrm{T}})}{r} \rfloor_q. \end{cases} \tag{29}$$

Based on $[\![\alpha_i]\!]$, $[\![\alpha_j]\!]$ and $[\![E_i]\!]$, $[\![b_1]\!]$ and $[\![b_2]\!]$ are separately obtained by

$$\begin{aligned}[\![b_1]\!] =&[\![b]\!] - [\![E_i]\!] - [\![y_i K_{ii}(\alpha_i - \alpha_i^{\text{old}})]\!] \\ &- \underset{j}{[\![}y_j K_{ij}(\alpha_j - \alpha^{\text{old}})]\!],\end{aligned} \tag{30}$$

and

$$\begin{aligned}[\![b_2]\!] =&[\![b]\!] - [\![E_j]\!] - [\![y_i K_{ij}(\alpha_i - \alpha_i^{\text{old}})]\!] \\ &- \underset{j}{[\![}y_j K_{jj}(\alpha_j - \alpha^{\text{old}})]\!].\end{aligned} \tag{31}$$

Here, we have

$$\begin{cases} [\![y_i K_{ii}]\!] = \frac{1}{m}\lceil \frac{\text{vec}([\![y_i]\!][\![K_{ii}]\!]^{\text{T}})}{r} \rfloor_q \\ [\![y_i K_{ii}(\alpha_i - \alpha_i^{\text{old}})]\!] = \frac{1}{m}\lceil \frac{\text{vec}([\![y_i K_{ii}]\!]([\![\alpha_i]\!] - [\![\alpha_i^{\text{old}}]\!])^{\text{T}})}{r} \rfloor_q \\ [\![y_j K_{ij}]\!] = \frac{1}{m}\lceil \frac{\text{vec}([\![y_j]\!][\![K_{ij}]\!]^{\text{T}})}{r} \rfloor_q \\ [\![y_j K_{ij}(\alpha_j - \alpha_j^{\text{old}})]\!] = \frac{1}{m}\lceil \frac{\text{vec}([\![y_j K_{ij}]\!]([\![\alpha_j]\!] - [\![\alpha_j^{\text{old}}]\!])^{\text{T}})}{r} \rfloor_q \end{cases} \tag{32}$$

Based on the analysis above, Algorithm 2 details the implementation of the SMO algorithm in ED (SMO-ED). First, H initializes the parameters and sends them to CS_A after encrypting them with PK_A. For convenience, selecting the first variable is frequently determined by judging the samples in turn. Specifically, for $i \in N$, CS_A calculates $[\![E_i]\!]$ and sends it to H for KKT condition violation judgment. If it violates the KKT condition, it is kept as the i-th variable of the optimization objective. Once the i-th variable is determined, any different data sample can be randomly selected as the j-th variable in line 11. Thus, H calculates the upper and lower bounds (i.e., H and L) based on the two variables selected. Based on the selected variables i and j, the optimization procedure with respect to them can be conducted. CS_A computes $[\![E_j]\!]$, $[\![\eta]\!]$ and saves $[\![\alpha_i]\!]$, $[\![\alpha_j]\!]$ without optimization (i.e., $[\![\alpha_i^{\text{old}}]\!]$, $[\![\alpha_j^{\text{old}}]\!]$). Then, it sends $[\![E_j]\!]$, $[\![\alpha_i^{\text{old}}]\!]$, $[\![\alpha_j^{\text{old}}]\!]$ and $[\![\eta]\!]$ to H. Since VHE operates on vectors, H is required to compute $[\![\frac{1}{\eta}]\!]$. Therefore, upon receiving $[\![\eta]\!]$, H decrypts it, calculates $\frac{1}{\eta}$, and then sends $[\![\frac{1}{\eta}]\!]$ (encrypted with PK_A) to CS_A. After receiving $[\![\frac{1}{\eta}]\!]$, CS_A calculates $[\![\alpha_j]\!]$ and returns it to H for the following pruning based on (20). Once getting the pruned result, CS_A computes $[\![\alpha_i]\!]$, and calculates $[\![b_1]\!]$ and $[\![b_2]\!]$ based on $[\![\alpha_i]\!]$ and $[\![\alpha_j]\!]$. After the computation is completed, CS_A sends the results to H who will decrypt and judge to obtain b. Further, H sends $[\![b]\!]$ back to CS_A for the next round of optimization until the termination condition is satisfied.

SVC Model Training with VHE. To collaboratively train the SVC model in ED, based on VHE, the whole procedure detailed in Algorithm 3 consists of three critical works, i.e., data processing, solving the dual problem, and model parameters calculations.

Algorithm 2 SMO-ED

Input: Dataset $[\![D]\!]$, penalty parameter C, tolerance tol
Output: $[\![\boldsymbol{\alpha}]\!]$, $[\![b]\!]$

H:
1: Initialize $\boldsymbol{\alpha}$ and b
2: $[\![\boldsymbol{\alpha}]\!] = \text{Enc}(PK_\text{H}, \boldsymbol{\alpha})$
3: $[\![b]\!] = \text{Enc}(PK_\text{H}, b)$
4: $\text{H} \rightarrow \text{CS}_\text{A}$: $[\![\boldsymbol{\alpha}]\!]$ and $[\![b]\!]$
5: **while** true **do**
6: **for** $i = 1, 2, \ldots, N$ **do**

CS_A:
7: calculate $[\![E_i]\!]$ following (24)
8: $\text{CS}_\text{A} \rightarrow \text{H}$: $[\![E_i]\!]$

H:
9: $E_i = \text{Dec}(SK_\text{H}, [\![E_i]\!])$
10: judge whether $(y_i E_i < -\text{tol})\,\&\&\,(\alpha_i < \text{C})$ or $(y_i E_i > \text{tol})\,\&\&\,(\alpha_i > 0)$
11: randomly select $j \neq i$
12: calculate L and H

CS_A:
13: calculate $[\![E_j]\!]$ and $[\![\eta]\!]$
14: $[\![\alpha_i^{\text{old}}]\!] \leftarrow [\![\alpha_i]\!]$, $[\![\alpha_j^{\text{old}}]\!] \leftarrow [\![\alpha_j]\!]$
15: $\text{CS}_\text{A} \rightarrow \text{H}$: $[\![E_j]\!]||[\![\alpha_i^{\text{old}}]\!]||[\![\alpha_j^{\text{old}}]\!]||[\![\eta]\!]$

H:
16: $\eta = \text{Dec}(SK_\text{H}, [\![\eta]\!])$
17: $[\![\frac{1}{\eta}]\!] = \text{Enc}\left(PK_\text{H}, \frac{1}{\eta}\right)$
18: $\text{H} \rightarrow \text{CS}_\text{A}$: $[\![\frac{1}{\eta}]\!]$

CS_A:
19: calculate $[\![\alpha_j]\!]$ following (26)
20: $\text{CS}_\text{A} \rightarrow \text{H}$: $[\![\alpha_j]\!]$

H:
21: $\alpha_j = \text{Dec}\left(SK_\text{H}, [\![\alpha_j]\!]\right)$
22: prune α_j following (20)
23: $[\![\alpha_j]\!] = \text{Enc}\left(PK_\text{H}, \alpha_j\right)$
24: $\text{H} \rightarrow \text{CS}_\text{A}$: $[\![\alpha_j]\!]$

CS_A:
25: calculate $[\![\alpha_i]\!]$, $[\![b_1]\!]$ and $[\![b_2]\!]$
26: $\text{CS}_\text{A} \rightarrow \text{H}$: $[\![\alpha_i]\!]||[\![b_1]\!]||[\![b_2]\!]$

H:
27: $(\alpha_i, b_1, b_2) = \text{Dec}(SK_\text{H}, [\![\alpha_i]\!]||[\![b_1]\!]||[\![b_2]\!])$
28: obtain b following (21)
29: $[\![b]\!] = \text{Enc}(PK_\text{H}, b)$
30: $\text{H} \rightarrow \text{CS}_\text{A}$: $[\![b]\!]$
31: **end for**
32: **end while**

- Data processing. H first performs matrix perturbation (that is, for an m-dimensional physical characteristics vector, multiply it by an m-dimensional orthogonal matrix $\mathbf{Q}$) and rounding processing on its stored data, then encrypts the processed data block by block and sends it to CS_A. In addition, H calculates the key-switching

matrix $\mathbf{M}$ based on the key-switching technique and sends it to CS_A as well.

- Solving the dual problem. After receiving the encrypted data and $\mathbf{M}$, based on the weighted inner product calculation in 3.1, CS_A quickly calculates the kernel function of SVM and cooperates with H to solve the dual problem by invoking SMO-ED.
- Model parameters calculations. Given $[\![\alpha]\!]$ and $[\![b]\!]$, H can easily calculate $[\![\mathbf{w}]\!]$ which, together with $[\![b]\!]$, will be outsourced to CS_A for online diagnosis.

Algorithm 3 SVC-VHE

Input: Dataset D, penalty parameter C, tolerance tol
Output: $[\![\mathbf{w}]\!]$, $[\![b]\!]$
H:
1: Generate an orthogonal matrix $\mathbf{Q}$, then
2: $\mathbf{x}' = \lceil \mathbf{Q}\mathbf{x} \rfloor_q$
3: $D' \leftarrow D' \cup \mathbf{x}'$
4: $[\![D']\!] = \mathrm{Enc}(PK_H, D')$
5: calculates $\mathbf{M}$ following (4)
6: H $\rightarrow$ CS_A: $[\![D']\!]$ and $\mathbf{M}$
CS_A:
7: calculates $[\![\mathbf{K}]\!]$ based on $[\![D']\!]$ following (10)
CS_A and H:
8: $[\![\alpha]\!], [\![b]\!] \leftarrow$ SMO-ED($[\![D']\!]$, C, tol)
H:
9: obtain $[\![\mathbf{w}]\!]$ following (12)
10: H$\rightarrow$ CS_A: $[\![\mathbf{w}]\!]$ and $[\![b]\!]$

Given $[\![\mathbf{w}]\!]$ and $[\![b]\!]$, the decision function can be constructed following (14). Meanwhile, the typical one-vs-rest (OVR) strategy is suggested by CPPOD to train multiple classification hyperplanes. As the expected medical diagnosis model, they will be securely outsourced to CS_A.

5.3. Online diagnosis

In the online diagnosis phase, communications occur among Us, H, and CS_A. Before submitting a request, Us should pass the authentication of CS_B. Following the privacy-preserving data collection algorithm (1), Us thus upload the physical characteristics $\mathbf{x}$ protected by $[\![K_S']\!]$ and PK_H for online diagnosis. For correct medical diagnosis, CPPOD suggests that the hospital has the highest authority. To regulate online diagnosis, the hospital, as the only owner of the model, can not be bypassed without authorization. Meanwhile, the hospital's participation should be lightweight for efficiency. Towards these considerations, Algorithm 4 presents the online diagnosis (OD) provided by CS_A with the outsourced SVC model from H.

Notice that the pathological data is valuable to the hospital. So, in OD, the request data $\mathbf{x}$ from Us will be stored in$D$ (for further model optimization) and sent to CS_A for analysis in ED after being encrypted. After receiving $[\![\mathbf{x}']\!]$, CS_A calculates the prediction values $[\![R]\!] = [\![(R_1, R_2, \cdots, R_z)]\!]$ and returns them to the hospital H. Then, H decrypts $[\![R]\!]$ and get the clinical diagnosis result d with a simple voting strategy based on $g(\mathbf{x})$. Besides, H sends $d' = d \oplus Rd$ to Us for security. For the sake of simplicity, we omit the negotiation for a random number Rd between H and Us since many practical key negotiation protocols (such as Diffie-Hellman key exchange) can be found in literature[35]. Apparently, H contributes to each diagnosis decision, and CS_A can not give a reasonable $[\![d']\!]$ without H. Even though CS_A conducts a replay attack by randomly sending a used one to Us, it will be picked out for incorrect Rd. Consequently, H can control each online diagnosis service offered by CS_A at the cost of lightweight operations in lines 9-12. That is, CS_A can not provide online diagnosis service without authorization from H.

Algorithm 4 OD - online diagnosis

Input: the encrypted physical characteristics $[\![\mathbf{x}]\!]$
Output: the clinical diagnosis result d
Us:
1: Us $\rightarrow$ H: $[\![\mathbf{x}]\!]$ and $[\![K_S']\!]$ (Data collection by Algorithm 1)
H:
2: $\mathbf{x}' = \lceil \mathbf{Q}\mathbf{x} \rfloor_q$
3: $[\![\mathbf{x}']\!] = \mathrm{Enc}(PK_H, \mathbf{x}')$
4: H $\rightarrow$ CS_A: $[\![\mathbf{x}']\!]$
CS_A:
5: **for** $z \in [1, N_{\mathrm{class}}]$ **do**
6: $\quad [\![R_z]\!] = \lceil \frac{\mathrm{vec}([\![\mathbf{w}]\!][\![\mathbf{x}']\!]^{\mathrm{T}})}{\mathrm{r}} \rfloor_{\mathrm{q}} + [\![b]\!]$
7: **end for**
8: $CS_A \rightarrow$ H: $[\![R]\!]$
H:
9: $R \leftarrow \mathrm{Dec}(SK_H, [\![R]\!])$
10: vote for d, then calculates $d' = d \oplus Rd$
11: $[\![d']\!] = \mathrm{Enc}(PK_U, d')$
12: H$\rightarrow$ Us: $[\![d']\!]$
Us:
13: $d' = \mathrm{Dec}(SK_U, [\![d']\!])$
14: $d = d' \oplus Rd$.

5.4. Work Mode of CPPOD

By introducing the phases above, i.e., data collection, model training, and online diagnosis, Figure.2 gives the flow diagram of CPPOD. For ease of distinction, three phases are highlighted by different backgrounds. Even in different phases, arrows starting from the same

end prefer the same style, i.e., the long dotted line representing communication launched by Us. The only double-sided arrow denotes a number of exchanges between H and CS_A due to iterations in SMO-ED. Eventually, as discussed by[36], a limited number of iterations can also give a reasonable performance. During the implementation of CPPOD, all the crucial information is encrypted. For clarity, we simplify the key distribution protocol by using $[\![\cdot]\!]$ to denote protection through encryption with the target's public key or signature with CS_B's private key. That is, CPPOD does not introduce any limitation over the key distribution.

6. SECURITY ANALYSIS

In this section, we prove the security of the proposed scheme from two aspects. Firstly, we demonstrate the correctness of the protocol. Secondly, we study the issue of privacy protection under the framework based on theoretical analysis.

6.1. Correctness

To ensure data security, we pre-processed the data before submitting it to CS_A. In order to verify the correctness of the results after data pre-processing, we will carry out theoretical proof.

Consider two sample vectors $\mathbf{x}_1$ and $\mathbf{x}_2$, the corresponding ciphertexts are $\mathbf{c}_1$ and $\mathbf{c}_2$, respectively. These ciphertexts can be decrypted using the key $\mathbf{S}$. More specifically,

$$\mathbf{S}\mathbf{c}_1 = q\mathbf{k}_1 + r\mathbf{x}_1 + \mathbf{e}_1, \tag{33}$$

$$\mathbf{S}\mathbf{c}_2 = q\mathbf{k}_2 + r\mathbf{x}_2 + \mathbf{e}_2. \tag{34}$$

Following the weighted inner products (10), we get

$$\lceil \mathrm{vec}\left(\mathbf{S}^{\mathrm{T}}\mathbf{S}\right)^{\mathrm{T}} \cdot \frac{\mathrm{vec}\left(\mathbf{c}_1\mathbf{c}_2^{\mathrm{T}}\right)}{r} \rfloor_q = q\mathbf{k} + r\left(\mathbf{x}_1^{\mathrm{T}}\mathbf{x}_2\right) + \mathbf{e} \tag{35}$$

Given an orthogonal matrix $\mathbf{Q}$ satisfying $\mathbf{Q}^T\mathbf{Q} = \mathbf{Q}\mathbf{Q}^T = \mathbf{I}$, we perturbs $\mathbf{x}_1$ and $\mathbf{x}_2$ to get $\mathbf{x}_1' = \mathbf{Q}\mathbf{x}_1$ and $\mathbf{x}_2' = \mathbf{Q}\mathbf{x}_2$. On the basis of VHE encryption, the ciphertext $\mathbf{c}_1'$ and $\mathbf{c}_2'$ can be separately calculated with the key $\mathbf{S}'$ as follows

$$\mathbf{S}'\mathbf{c}_1' = q\mathbf{k}_1' + r\mathbf{x}_1' + \mathbf{e}_1', \tag{36}$$

$$\mathbf{S}'\mathbf{c}_2' = q\mathbf{k}_2' + r\mathbf{x}_2' + \mathbf{e}_2', \tag{37}$$

where $\mathbf{S}' = \mathbf{Q}\mathbf{S}$. So, we have

$$\lceil \frac{\mathrm{vec}\left(\mathbf{S}^{\mathrm{T}}\mathbf{S}\right)^{\mathrm{T}} \cdot \mathrm{vec}\left(\mathbf{c}_1'\mathbf{c}_2'^{\mathrm{T}}\right)}{r} \rfloor_q = q\mathbf{k}' + r\left(\mathbf{x}_1^{\mathrm{T}}\mathbf{x}_2\right) + \mathbf{e}' \tag{38}$$

Proof. First, by introducing (10), we get

$$\lceil \frac{\mathrm{vec}\left(\mathbf{S}'^{\mathrm{T}}\mathbf{S}'\right)^{\mathrm{T}} \cdot \mathrm{vec}\left(\mathbf{c}_1'\mathbf{c}_2'^{\mathrm{T}}\right)}{r} \rfloor_q = q\mathbf{k}' + r\left(\mathbf{x}_1'^{\mathrm{T}}\mathbf{x}_2'\right) + \mathbf{e}'. \tag{39}$$

Since $\mathbf{Q}^T\mathbf{Q} = \mathbf{Q}\mathbf{Q}^T = \mathbf{I}$, we have

$$\begin{aligned} \mathbf{x}_1'^T\mathbf{x}_2' &= (\mathbf{Q}\mathbf{x}_1)^{\mathrm{T}}(\mathbf{Q}\mathbf{x}_2) \\ &= \mathbf{x}_1^T\mathbf{Q}^T\mathbf{Q}\mathbf{x}_2 \\ &= \mathbf{x}_1^T\mathbf{I}\mathbf{x}_2 \\ &= \mathbf{x}_1^T\mathbf{x}_2, \end{aligned} \tag{40}$$

and

$$\begin{aligned} \mathbf{S}'^{\mathrm{T}}\mathbf{S}' &= (\mathbf{Q}\mathbf{S})^{\mathrm{T}}(\mathbf{Q}\mathbf{S}) \\ &= \mathbf{S}^T\mathbf{Q}^T\mathbf{Q}\mathbf{S} \\ &= \mathbf{S}^T\mathbf{I}\,\mathbf{S} \\ &= \mathbf{S}^T\mathbf{S}. \end{aligned} \tag{41}$$

Based on (40) and (41), thus (39) becomes

$$\lceil \frac{\mathrm{vec}\left(\mathbf{S}^{\mathrm{T}}\mathbf{S}\right)^{\mathrm{T}} \cdot \mathrm{vec}\left(\mathbf{c}_1'\mathbf{c}_2'^{\mathrm{T}}\right)}{r} \rfloor_q = q\mathbf{k}' + r\left(\mathbf{x}_1^{\mathrm{T}}\mathbf{x}_2\right) + \mathbf{e}'. \tag{42}$$

□

According to (42), we find that by decrypting ciphertext $\lceil \frac{\mathrm{vec}(\mathbf{c}_1'\mathbf{c}_2'^{\mathrm{T}})}{r} \rfloor_q$ with the secret key $\mathrm{vec}(\mathbf{S}^{\mathrm{T}}\mathbf{S})^{\mathrm{T}}$, the inner product of $\mathbf{x}_1$ and $\mathbf{x}_2$ can be obtained. So, the perturbation by multiplying $\mathbf{Q}$ does not affect the kernel function calculation, as well as the calculation of $\boldsymbol{\alpha}$ and b.

Since $\mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i$, however, $\mathbf{w}$ will change to $\mathbf{Q}\mathbf{w}$ when $\mathbf{x}_i$ is replaced by $\mathbf{Q}\mathbf{x}_i$, i.e., $\mathbf{Q}\mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i (\mathbf{Q}\mathbf{x}_i)$. Therefore, when we formulate the classification hyperplane in ED, we will get the ciphertext $\mathbf{c}_w'$ of $\mathbf{Q}\mathbf{w}$ instead of $\mathbf{c}_w$ with respect to $\mathbf{w}$. Once a new physical characteristics vector $\mathbf{x}$ arrives, it will be perturbed and changed to $\mathbf{Q}\mathbf{x}$. Fortunately, according to (42), the diagnosis result will not be affected when $\mathbf{w}^T\mathbf{x}$ is calculated in ED.

6.2. Privacy Protection

We discuss the privacy of physical characteristics and model data. Our primary concern is that cloud servers do not know sensitive information.

First, in the model training phase, we preprocess the original data samples and encrypt them with the hospital's public key PK_{H}. Since the corresponding private key SK_{H} can only be held by the hospital, the outsourced dataset $[\![D]\!]$ can not be decrypted by the cloud server.

Secondly, an orthogonal matrix $\mathbf{Q} \in R^{m\times m}$ generated randomly is employed to perturb the original data. Thus, even though the server gets several data samples of D, to leak the original data, it must evaluate the matrix equivalent to constructing m^2 linear equations to solve the unknown variables of $\mathbf{Q}$. Unfortunately, this is a hard problem for the cloud server since different vector tuples in D are independent. Additionally, due

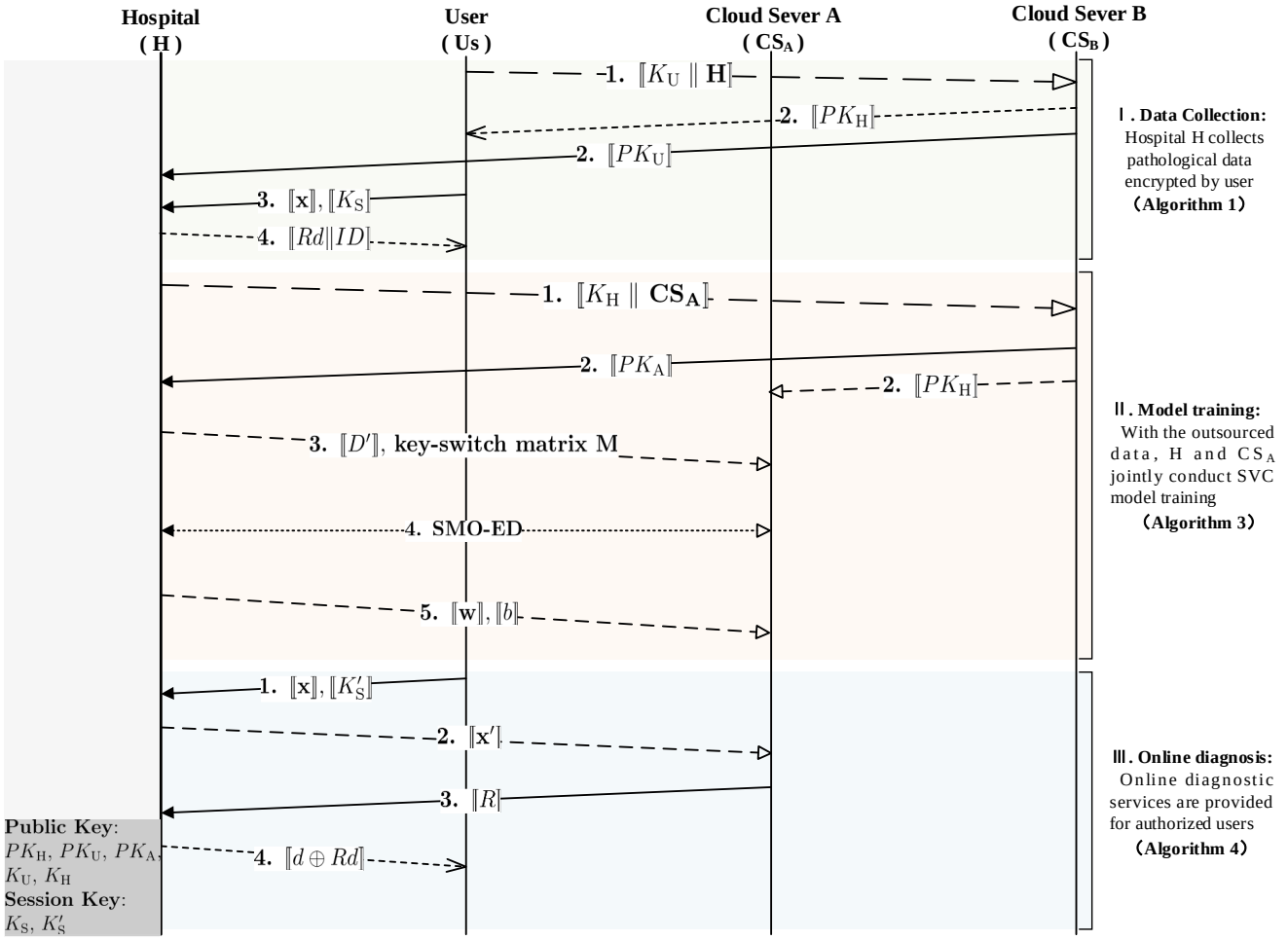


Figure 2. The flow diagram of the proposed CPPOD scheme

to the disturbance of the data, there will be a gap (as discussed in Section 6) between the final model and the original model. Therefore, the privacy of the SVC model can be guaranteed.

7. PERFORMANCE ANALYSIS

In this section, we conduct a performance analysis of the proposed CPPOD scheme in terms of time and communication complexity.

7.1. Time Complexity

Suppose D contains N data samples, N_{sv} is the number of support vectors, N_{class} is the number of classes, T_k is the time consumption for key generation, and a unit time is required for once encryption or decryption. So, it takes $O(k)$ to perform an encryption or decryption operation on k data samples in the form vector. Once receiving a communication request, CS_B will conduct an identity authentication for the requester that costs a fixed amount of time. Table 3 lists the time complexity of each participant at different phases.

Data Collection Phase. The data collection phase involves specific tasks done by the hospital H and any user of Us. For Us, all the data samples should be encrypted by a newly generated session secret key. Therefore, if a user wants to upload N_{Us} data samples, the time cost by the user is $O(N_{Us} + T_k)$. For H, it will decrypt the user's data and then generate an ID and random number Rd for the user. We omit the time of generating the ID and random number, so the total time complexity of the hospital is $O(N_{Us})$.

Model Training Phase. The model training phase is finished between the hospital H and the cloud server CS_A. Before outsourcing data to CS_A, H takes $O(N)$ to encrypt D, $O(nn')$ to generate a key-switch matrix $\mathbf{M}$, and $O(m^2)$ to generate a random orthogonal matrix $\mathbf{Q}$, respectively. Once entering the iteration process of SMO-ED, H only needs to decrypt intermediate data and make simple judgments. Let ζ be the essential number of iterations, and each iteration requires p decryptions and encryptions on average. The hospital thus spends $O(\zeta p)$. Notice that p is proportional to the

Table 3. Time complexities of the participators except the key server CS_B in CPPOD scheme

Phase	Users (Us)	Hospital (H)	Cloud Server (CS_A)
Data Collection	$O(N_{Us} + T_k)$	$O(N_{Us})$	–
Model Training	—	$O\left(N_{\text{class}}(\zeta p + T_w) + N + nn' + m^2\right)$	$O\left(N_{\text{class}}\left(\left(N^2 + t\right)n^3 + un^2\right)\right)$
Online Diagnosis	$O\left(N_{U_i} + T_{\text{XOR}}\right)$	$O\left(N_{U_i}(N_{\text{class}} + 1) + T_{\text{XOR}}\right)$	$O\left(N_{U_i}N_{\text{class}}\left(n^3 + n^2\right)\right)$

number of sample pairs violating the KKT condition. Besides, the hospital needs to calculate the final model parameters following (12) that can be denoted by $O(T_w)$. Therefore, for a N_{class} classes problem, the total time complexity for the hospital H is $O(N_{\text{class}}(\zeta p + T_w) + N + nn' + m^2)$.

For CS_A, it takes $O(N^2n^3)$ to obtain the kernel function matrix since an inner product consumes $O(n^3)$. During the iteration process, CS_A also performs calculations, mostly for the inner product and addition operations on vectors. If each iteration requires t times the inner product and u times the addition calculation, a total of $O(tn^3 + un^2)$ is spent. Thus, CS_A consumes $O(N_{\text{class}}((N^2 + t)n^3 + un^2))$ when training N_{class} sub-models in ED.

Online Diagnosis Phase. The phase of online diagnosis requires participation from Us, H, and CS_A. Us only encrypt their data, decrypt the final result, and perform an XOR calculation. If an user submits N_{U_i} query data samples and each XOR calculation costs $O(T_{\text{XOR}})$, then the user consumes $O(N_{U_i} + T_{\text{XOR}})$ in total. For H, it firstly costs $O(N_{U_i})$ to decrypt and encrypt the query data. Meanwhile, it will also require $O(N_{U_i}N_{\text{class}})$ to make decision based on N_{U_i} intermediate diagnosis results $[\![R]\!]$ and send $[\![d']\!]$ to the user. That means H has to consume $O(N_{U_i}(N_{\text{class}} + 1) + T_{\text{XOR}})$. Finally, for CS_A, the inner product and addition operations in ED cause a total time complexity of $O(N_{U_i}N_{\text{class}}(n^3 + n^2))$.

7.2. Communication Complexity

Regarding the communication complexity of the proposed CPPOD scheme, the transmission of physical characteristics $[\![\mathbf{x}]\!]$, dataset $[\![D]\!]$, key-switch matrix $\mathbf{M}$, parameters in SMO-ED, model parameters $[\![\mathbf{w}]\!]$, $[\![b]\!]$, intermediate diagnosis results $[\![R]\!]$ and the final result $[\![d \oplus Rd]\!]$ are the main causes of network bandwidth consumption. Since an m-dimensional vector will become a n-dimensional vector after encryption. Generally, the numeric type of double supplies enough precision for computation, which requires 8 Bytes for each item. So, the size of an encrypted vector is $8n$ bytes. For clarity, we show the bandwidth consumed at each stage for a single round. For the data collection phase, the user uploads N_{Us} physical characteristics $[\![\mathbf{x}]\!]$ consume $8nN_{\text{Us}}$ bytes of bandwidth. For the model training phase, H takes $8n(N + n')$ bytes to send the medical dataset $[\![D]\!]$ and key-switch matrix $\mathbf{M}$. In the iteration process, each iteration requires an average of p times of data transmission, which is equal to the number of encryption and decryption operations carried out by the hospital. Since ζ iterations are assumed, a total of $8np\zeta N_{\text{class}}$ bytes is consumed when there are N_{class} classes in the iteration process. Moreover, after the hospital calculates the model parameters, it outsources $[\![\mathbf{w}]\!]$, $[\![b]\!]$ to the server, which consumes $16nN_{\text{class}}$ bytes. For the online diagnosis phase, $8nN_{u_i}$ bytes are required for the user to submit N_{u_i} query data samples, which is equal to the bandwidth consumed by the hospital to transmit the pre-processed data samples. In addition, $8nN_{\text{class}}N_{U_i}$ bytes are required when the server sends intermediate diagnosis result $[\![R]\!]$ to the hospital, and $8n$ bytes are required to send the final result to the user.

8. EXPERIMENTAL RESULTS

8.1. Experimental Setup

To conduct effectiveness and efficiency analysis, we have implemented the proposed CPPOD by mixed programming with MATLAB and C++ with NTL library version 9.6.0[1]. For the sake of simplicity, programs corresponding to the cloud server, the hospital, and the client are modeled as different threads of a single program, which passes data or parameters to each other following the rules shown in Figure.2. Meanwhile, we conducted all the experiments on a computer with Quad-Core 2.30 GHz CPUs and 40 GB main memory running on Windows 10-X64.

Due to different model architectures adopted by the existing scheme, as well as distinguishing implementations of the involved phases, in this section, our experiments will concentrate on the effectiveness analysis in ED compared with linear SVC in PD and efficiency checks of whether the four-party scheme avoiding imposing pricey computation on both the hospital and user.

[1]https://libntl.org/

8.2. Experimental Dataset

For the experiments, we consider three typical medical datasets with multiple classes provided by UC Irvine Machine Learning Repository(UCI)[2], i.e., Breast Cancer Wisconsin (Original)(BCW) [37], Maternal Health Risk Data (Maternal) [38], and HCV data [39]. Table 4 shows the statistical information of these datasets.

BCW contains 699 data samples with 9-dimensional integer attributes separately represent Clump_thickness, Uniformity_of_cell_size, Uniformity_of_cell_shape, Marginal_adhesion, Single_epithelial_cell_size, Bare_nuclei, Bland_chromatin, Normal_nucleoli, and Mitoses. All the data samples are divided into benign and malignant classes with imbalanced distribution. Maternal contains 1014 data samples from three risk levels: low risk, mid risk, and high risk. The amount of data in these three categories is slightly unbalanced. Different from BCW, six attributes for each data sample in Maternal correspond to age, SystolicBP, DiastolicBP, BS, BodyTemp, and HeartRate. The final data HCV has 615 samples in five highly unbalanced classes: Blood Donor, suspect Blood Donor, Hepatitis, Fibrosis, and Cirrhosis. Twelve attributes are Age, Sex, ALB, ALP, AST, BIL, CHE, CHOL, CREA, CGT, PROT, and ALT. Different from BCW and Maternal, these attributes are mixed with integer, binary, and real variables. These sensitive attributes in medical datasets raise privacy concerns. Therefore, towards online diagnosis in ED, the essential preprocessing with multiplier is employed as described in Section 8.4.

For fairness and objectivity, all the datasets are grouped in a ratio of 7 to 3, of which 70 percent is used as the training data for SVC model training, and the remaining 30 percent is employed as the testing data submitted by users in terms of online diagnosis queries. Furthermore, data for each experiment is randomly scrambled such that we can conduct at least ten rounds of experiments to get the final evaluation on average. To evaluate the proposed CPPOD scheme, we conduct several series of experiments in both PD and ED described in the following sections.

8.3. Experiments in the Plain Domain

In this section, we conduct experiments in PD to check the validity of CPPOD following the four-party flow diagram. Both the average accuracies of online diagnosis and time-consumptions for each party over the three medical datasets in PD are considered.

Benchmark Results for Classification Accuracy. To facilitate the experiment, we adopt the OVR strategy for these multiple classification problems and take Liblinear, which is a typical implementation of the linear SVC by [40] as the baseline. By setting C to 1, columns 2-3 of Table 5 illustrate the benchmark results separately achieved by Liblinear and the proposed CPPOD in PD. CPPOD in PD reaches diagnosis accuracies of 96.07%, 46.63%, and 86.63% on BCW, Maternal, and HCV, respectively. Notice that CPPOD in PD performs even better than Liblinear on HCV data. Therefore, these results confirm that CPPOD in PD can achieve comparable performance with Liblinear even though all the datasets have been processed to match the requirement of VHE.

Time-Consumption Required by The Hospital. In PD, the model training phase can only be conducted by the hospital. Until the online diagnosis phase, the user participates in submitting a diagnosis query that can be omitted in time-consumption analysis. For each dataset, therefore, we separately measure the time the hospital consumes in these two phases. On BCW, the hospital costs 0.024s and 0.002s to finish the model training and online diagnosis, respectively. On Maternal, these two phases for the hospital are separately 0.265s and 0.002s. On HCV data, they become 0.377s and 0.002s. Due to fast computations of the linear kernel, classification on the 30% testing data samples is almost indistinguishable.

8.4. Experiments in the Encrypted Domain

In this section, we conduct a series of experiments to verify the validity of CPPOD in terms of effectiveness and time consumption in ED. As shown in Algorithm 4, data collection (Algorithm 1) can be a part of the online diagnosis service. Therefore, for the whole system of CPPOD, the time-consumption measurement can be simplified by concentrating on the main occupation phases, model training and online diagnosis for clarity. In these two phases, each participant's costs will be separately evaluated.

Effectiveness Measurement in ED. As mentioned in Section 3.1, VHE is targeted at integer vectors. Towards better conducting the subsequent experiments, we introduce a scaling factor $mul = 10^4$ following [25, 36] for the dataset with hybrid data types, e.g., HCV data. Since mul is a linear scalar, its impacts can be restricted or eliminated along with the procedure of CPPOD. Or, a small yet practicable iteration number like what was discussed in [36] can also avoid unnecessary operations in ED. However, we prefer the former for clarity. Column 4 of Table 5 lists the average accuracy obtained by CPPOD over encrypted datasets of BCW, Maternal, and HCV data. CPPOD in ED has very close accuracies with that achieved by CPPOD in PD. The random partition of training and testing data brings tiny differences between CPPOD

[2]https://archive.ics.uci.edu/

Table 4. The Details of the Datasets

Dataset	No. Samples	No. Attributes	No. Classes	Classes and the number of each class				
BCW	699	9	2	benign: 458	malignant: 241			
Maternal	1014	6	3	low risk: 406	mid risk: 336	high risk: 272		
HCV data	615	12	5	Blood Donor: 533	suspect Blood Donor: 7	Hepatitis: 24	Fibrosis: 21	Cirrhosis: 30

Table 5. Accuracies Achieved by Liblinear[40], and CPPOD in PD and ED

Dataset	Liblinear[40]	CPPOD (in PD)	CPPOD (in ED)
BCW	96.19%	96.07%	95.95%
Maternal	51.64%	46.63%	46.37%
HCV data	86.41%	86.63%	84.78%

in PD and CPPOD in ED. Meanwhile, CPPOD in PD and CPPOD in ED achieve comparable results with Liblinear[40] on BCW and HCV data, but there is about 10.20% reduction on the linear inseparable Maternal. Therefore, classification over encrypted data is not suggested for low-dimensional and linear inseparable data due to the accumulated loss of data precision. However, this is not a challenging issue. On the one hand, we can increase the scaling factor to utilize VHE's advantage of supporting vector computation. On the other hand, we can also change the proposed CPPOD to support nonlinear separable data by replacing the linear kernel function with a nonlinear one (e.g., radial basis function) and fine-tuning lines 7-9 of SMO-ED. Moreover, a pre-computation for kernel matrix in ED may also benefit model training for using both computation and storage advantages of the cloud server. As aforementioned, the proposed CPPOD can be well applied or extended to any classification problems with privacy concerns for its ability to handle encrypted data without noticeable classification accuracy loss.

Time-Consumptions for Each Participant in ED. We measure the time consumption in terms of the average running time required by each participant in the proposed scheme. Results are listed in Table 6, where "(Enc/Dec Ratio)" denotes the proportion of run-time cost by data encryption and decryption function in the corresponding algorithms, e.g., "$[\![\boldsymbol{\alpha}]\!] = \mathrm{Enc}(PK_{\mathrm{H}}, \boldsymbol{\alpha})$" in line 2 of SMO-ED. Just as HE operations are conducted by $\mathrm{CS_A}$, the major time consumptions of H and Us are data encryption and decryption, which make model training and online diagnosis over encrypted data practical. To make the results more intuitive, we also use Figure.3 to depict the proportion of time consumed by each participant in the two phases. Apparently, for privacy protection, both model training and online diagnosis consume much more than consumptions in PD, as discussed in Section 8.3. Even so, the role of $\mathrm{CS_A}$ affords the vast majority of computations, particularly in the model training phase. Since both $\mathrm{CS_A}$ and H run on the same platform, the observations are easy to get. Once we transfer $\mathrm{CS_A}$ to a cloud with sufficient computing resources, its time consumption will significantly reduce. Notice that Us and H have very similar time consumptions (< 2.0s) in the phase of online diagnosis that are acceptable for making medical diagnoses on about 185 to 305 samples. Therefore, for practical use, with the assistance of $\mathrm{CS_A}$, we can also move the capabilities of Us and H to a smart terminal once the model training is completed.

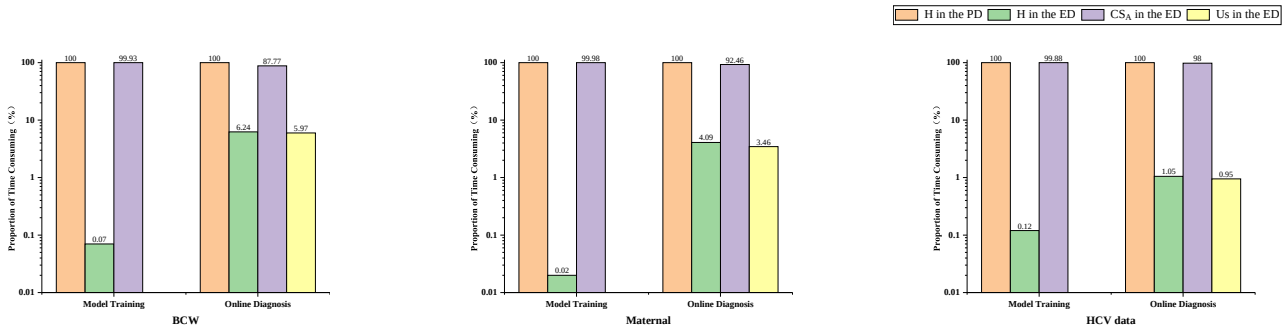
9. CONCLUSION

In this paper, a new privacy-preserving SVC scheme, namely CPPOD, is proposed to respond to privacy concerns in online medical diagnosis. Based on a four-party participation model architecture, it endows the hospital control ability over the online diagnosis at a tiny computation cost. Based on existing research, the core works of SVC can be optimized to achieve linear operations. Therefore, we present an SMO-ED solver to solve the classical dual problem (11) in ED by introducing VHE. Thus, the expected medical diagnosis model can be obtained by SVC-VHE over encrypted data. Further, the trained model can be outsourced to the computing cloud server, which offers online diagnosis service to users under the control and monitoring of the hospital. To provide a reasonable service, the procedure control and monitoring can not be bypassed. Furthermore, in the four-party architecture, the vast majority of computations are afforded by the computing cloud server. Theoretical analysis proves the correctness and security of CPPOD, and experimental results give evidence corresponding to effectiveness and efficiency. However, even though CPPOD can be easily extended to support nonlinear separable problems as discussed in Section 8.4, interactions between the computing cloud server and the hospital due to frequent iterations in SMO-ED require further control for energy consumption reduction. Therefore, improving the design of the dual problem solver with fundamental operations well-matching HE is valuable to be investigated in the future.

Acknowledgements. The authors would like to thank Prof. Xiaojun Wang (Dublin City University) for kind suggestion on the solution, and the Associate Editor and the anonymous reviewers for their constructive comments that greatly improved the quality of this manuscript. This work is supported by the National Natural Science Foundation of

Table 6. Runtime(s) of each phase on all the datasets in ED

Dataset	Model Training (s.)			Online Diagnosis (s.)			
	Total	H (Enc/Dec Ratio)	CS_A	Total	Us (Enc/Dec Ratio)	H (Enc/Dec Ratio)	CS_A
BCW	24024.3	16.5 (98.04%)	24007.8	17.2	1.0 (99.90%)	1.1 (99.71%)	15.1
Maternal	216668.6	52.3 (97.81%)	216615.9	18.5	0.6 (99.83%)	0.8 (98.39%)	17.1
HCV data	100318.4	122.9 (98.83%)	100195.1	177.4	1.7 (99.94%)	1.9 (99.45%)	173.8



(a) BCW (b) Maternal (c) HCV data

Figure 3. The time proportion consumed by each participant in model training and testing.

China under Grant No. 62162009 and 62101478, the Key Technologies R&D Program of He'nan Province under Grant No. 222102210048, the Key Research and Development in He'nan Province under Grant 22A520040, the Scientific Research Innovation Team of Xuchang University under Grant No. 2022CXTD003, and Innovation Scientists and Technicians Troop Construction Projects of Henan Province under Grant No. CXTD2017099.

References

[1] Tjoa, E. and Guan, C. (2021) A survey on explainable artificial intelligence (xai): Toward medical xai. *IEEE Trans Neural Netw Learn Syst* **32**(11): 4793–4813.

[2] Ganesan, M. and Sivakumar, N. (2019) Iot based heart disease prediction and diagnosis model for healthcare using machine learning models. In *2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN)* (Pondicherry, India): 1–5.

[3] Zhang, Y.D., Dong, Z., Wang, S.H., Yu, X., Yao, X., Zhou, Q., Hu, H. *et al.* (2020) Advances in multimodal data fusion in neuroimaging: Overview, challenges, and novel orientation. *Inf Fusion* **64**: 149–187.

[4] Liang, J., Qin, Z., Xiao, S., Ou, L. and Lin, X. (2021) Efficient and secure decision tree classification for cloud-assisted online diagnosis services. *IEEE Trans Dependable Secure Comput* **18**(4): 1632–1644.

[5] Zhou, X., Li, Y. and Liang, W. (2021) Cnn-rnn based intelligent recommendation for online medical pre-diagnosis support. *IEEE ACM T COMPUT BI* **18**(3): 912–921.

[6] Shi, B., Ye, H., Zheng, J., Zhu, Y., Heidari, A.A., Zheng, L., Chen, H. *et al.* (2021) Early recognition and discrimination of covid-19 severity using slime mould support vector machine for medical decision-making. *IEEE Access* **9**: 121996–122015.

[7] Zhu, H., Liu, X., Lu, R. and Li, H. (2017) Efficient and privacy-preserving online medical prediagnosis framework using nonlinear svm. *IEEE J Biomed Health Inform* **21**(3): 838–850.

[8] Wang, S.H., Govindaraj, V.V., Górriz, J.M., Zhang, X. and Zhang, Y.D. (2021) Covid-19 classification by fgcnet with deep feature fusion from graph convolutional network and convolutional neural network. *Inf Fusion* **67**: 208–229.

[9] Abawajy, J.H. and Hassan, M.M. (2017) Federated internet of things and cloud computing pervasive patient health monitoring system. *IEEE Commun Mag* **55**(1): 48–53.

[10] Wang, S.H., Nayak, D.R., Guttery, D.S., Zhang, X. and Zhang, Y.D. (2021) Covid-19 classification by ccshnet with deep fusion using transfer learning and discriminant correlation analysis. *Inf Fusion* **68**: 131–148.

[11] Liang, J., Qin, Z., Ni, J., Lin, X. and Shen, X. (2021) Practical and secure svm classification for cloud-based remote clinical decision services. *IEEE Trans Comput* **70**(10): 1612–1625.

[12] Chen, Y., Mao, Q., Wang, B., Duan, P., Zhang, B. and Hong, Z. (2022) Privacy-preserving multi-class support vector machine model on medical diagnosis. *IEEE J Biomed Health Inform* **26**(7): 3342–3353.

[13] Liang, J., Qin, Z., Xue, L., Lin, X. and Shen, X. (2021) Verifiable and secure svm classification for cloud-based health monitoring services. *IEEE Internet Things J* **8**(23): 17029–17042.

[14] Li, X., Zhu, Y., Wang, J., Liu, Z., Liu, Y. and Zhang, M. (2018) On the soundness and security of privacy-preserving svm for outsourcing data classification. *IEEE*

Trans Dependable Secure Comput **15**(5): 906–912.

[15] Liang, J., Qin, Z., Xue, L., Lin, X. and Shen, X. (2021) Efficient and privacy-preserving decision tree classification for health monitoring systems. *IEEE Internet Things J* **8**(16): 12528–12539.

[16] Liu, X., Deng, R.H., Choo, K.K.R. and Yang, Y. (2020) Privacy-preserving outsourced support vector machine design for secure drug discovery. *IEEE Trans. on Cloud Comput* **8**(2): 610–622.

[17] Bajard, J.C., Martins, P., Sousa, L. and Zucca, V. (2020) Improving the efficiency of svm classification with fhe. *IEEE Trans. Inf. Forensics Secur* **15**: 1709–1722.

[18] Jia, Q., Guo, L., Jin, Z. and Fang, Y. (2018) Preserving model privacy for machine learning in distributed systems. *IEEE Trans Parallel Distrib Syst* **29**(8): 1808–1822.

[19] Dong, C., Weng, J., Liu, J.N., Yang, A., Liu, Z., Yang, Y. and Ma, J. (2023) Maliciously secure and efficient large-scale genome-wide association study with multi-party computation. *IEEE Trans Dependable Secure Comput* **20**(2): 1243–1257.

[20] Qayyum, A., Qadir, J., Bilal, M. and Al-Fuqaha, A. (2021) Secure and robust machine learning for healthcare: A survey. *IEEE Rev Biomed Eng* **14**(7): 156–180.

[21] Liu, L., Chen, R., Liu, X., Su, J. and Qiao, L. (2020) Towards practical privacy-preserving decision tree training and evaluation in the cloud. *IEEE Trans. Inf. Forensics Secur* **15**: 2914–2929.

[22] Zheng, Y., Duan, H., Wang, C., Wang, R. and Nepal, S. (2022) Securely and efficiently outsourcing decision tree inference. *IEEE Trans Dependable Secure Comput* **19**(3): 1841–1855.

[23] Zhou, H. and Wornell, G. (2014) Efficient homomorphic encryption on integer vectors and its applications. In *2014 Information Theory and Applications Workshop (ITA)* (San Diego, CA, USA): 1–9.

[24] Yu, A., Lai, W.L. and Payor, J. (2015) Efficient integer vector homomorphic encryption (Massachusetts Institute of Technology). URL https://courses.csail.mit.edu/6.857/2015/files/yu-lai-payor.pdf.

[25] Rahulamathavan, Y., Veluru, S., Phan, R.C.W., Chambers, J.A. and Rajarajan, M. (2014) Privacy-preserving clinical decision support system using gaussian kernel-based classification. *IEEE J Biomed Health Inform* **18**(1): 56–66.

[26] Zhang, M., Song, W. and Zhang, J. (2022) A secure clinical diagnosis with privacy-preserving multiclass support vector machine in clouds. *IEEE Syst J* **16**(1): 67–78.

[27] Chen, H., Ünal, A.B., Akgün, M. and Pfeifer, N. (2020) Privacy-preserving svm on outsourced genomic data via secure multi-party computation. In *Proceedings of the Sixth International Workshop on Security and Privacy Analytics (IWSPA'20)* (New York, United States): 61–69.

[28] Wang, J., Wu, L., Wang, H., Choo, K.K.R. and He, D. (2021) An efficient and privacy-preserving outsourced support vector machine training for internet of medical things. *IEEE Internet Things J* **8**(1): 458–473.

[29] Xie, B., Xiang, T., Liao, X. and Wu, J. (2022) Achieving privacy-preserving online diagnosis with outsourced svm in internet of medical things environment. *IEEE Trans Dependable Secure Comput* **19**(6): 4113–4126.

[30] Wang, F., Zhu, H., Lu, R., Zheng, Y. and Li, H. (2022) Achieve efficient and privacy-preserving disease risk assessment over multi-outsourced vertical datasets. *IEEE Trans Dependable Secure Comput* **19**(3): 1492–1504.

[31] Zhang, M., Chen, Y. and Susilo, W. (2023) Decision tree evaluation on sensitive datasets for secure e-healthcare systems. *IEEE Trans Dependable Secure Comput* **20**(5): 3988–4001.

[32] Liu, L., Su, J., Liu, X., Chen, R., Huang, K., Deng, R.H. and Wang, X. (2019) Toward highly secure yet efficient knn classification scheme on outsourced cloud data. *IEEE Internet Things J.* **6**(6): 9841–9852.

[33] Gao, C.z., Li, J., Xia, S., Choo, K.K.R., Lou, W. and Dong, C. (2022) Mas-encryption and its applications in privacy-preserving classifiers. *IEEE Trans Knowl Data Eng* **34**(5): 2306–2323.

[34] Keerthi, S.S., Shevade, S.K., Bhattacharyya, C. and Murthy, K.R.K. (2001) Improvements to platt's smo algorithm for svm classifier design. *Neural Comput* **13**(3): 637–649.

[35] Xiong, H., Wu, Y. and Lu, Z. (2019) A survey of group key agreement protocols with constant rounds. *ACM Comput. Surv.* **52**(3): 57:1–32.

[36] Ping, Y., Hao, B., Hei, X., Wu, J. and Wang, B. (2020) Maximized privacy-preserving outsourcing on support vector clustering. *Electronics* **9**(1): 178:1–30.

[37] Wolberg, W. (1992), Breast cancer wisconsin (original), UCI Machine Learning Repository. DOI:https://doi.org/10.24432/C5HP4Z (accessed on 10 January 2023).

[38] Ahmed, M. (2023), Maternal Health Risk, UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5DP5D (accessed on 2 Febuary 2023).

[39] Lichtinghagen, R., Klawonn, F. and Hoffmann, G. (2020), Hcv data, UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5D612 (accessed on 15 Febuary 2023).

[40] Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R. and Lin, C.J. (2008) Liblinear: A library for large linear classification. *J. Mach. Learn. Res.* **9**: 1871–1874.