# Outsourced Privacy-Preserving SVM Classifier Model over Encrypted Data in IoT

Chen Wang[1], Yan Dong[1], Jiarun Li[1], Chen Chen[2], and Jian Xu[1,3](✉)

[1] Software College, Northeastern University, Shenyang 110169, China
xuj@mail.neu.edu.cn
[2] Northeastern University, Shenyang 110169, China
[3] State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China

**Abstract.** The Internet of Things (IoT) enables the development of cloud computing by combining machine learning (ML) and big data technologies. Frameworks supporting ML typically process and classification manufacturing data services via cloud-based technologies. In order to achieve secure and efficient data collection and application, we design, implement and evaluate a new system employing a SVM classifier model over encrypted data (SVMCM-ED) based on multi-key Fully Homomorphic Encryption (multi-key FHE) in IoT. We first propose a new scheme that uses the cloud server and edge node to jointly implement SVM classification over the ciphertext to ensure the security of data and classification model. Our system significantly transfers cloud processing to the edge, and safely uses the model obtained from the data analysis to implement SVM classification among multiple users. We further design secure protocols based on multi-key FHE, which supports multiple users and satisfies the semi-honest security model where cloud computing is outsourced. Our protocol requires no interaction from the data analysis and users during online secure classification. Performance evaluation demonstrates that our new system can securely train a logistic regression model of multiple users. Performance evaluation demonstrates that our new system can securely implement the SVM classification model of multiple users in IoT, as well as share communication and computation overhead of the cloud.

**Keywords:** Multi-key FHE · cloud-edge · privacy-preserving · Internet of Things (IoT) · SVM

## 1 Introduction

Classification is a very important data mining method [1,2]. The concept of classification is to learn classification functions or build classification models based on existing data. This function or model maps the data records in the database

Q. Jiang et al. (Eds.): SPNCE 2022, LNICST 496, pp. 82–94, 2023.
https://doi.org/10.1007/978-3-031-30623-5_6

to one of the given categories, which can be applied to data prediction. The basic model of support vector machine (SVM) is to find the best separation hyperplane in the feature space to maximize the separation of positive and negative samples on the training set. SVM is a supervised learning algorithm for binary classification. After introducing kernel method, support vector machine can also be used to solve nonlinear problems.

Machine learning has brought great benefits to various application fields of the Internet of Things (IoT) [3], and put forward higher requirements for computing and storage resources. On the one hand, the processing capacity of personal terminals in practical applications is extremely limited, and they need to process massive data with the help of a trusted third-party cloud server. At the same time, centralized storage and processing of large amounts of data in a single cloud server will cause serious bandwidth and power consumption problems. Edge nodes are introduced to reduce the bandwidth and computing burden of the cloud server [4,5]. On the other hand, data storage in the cloud server and edge nodes brings about data leakage [6,7]. If any private data is disclosed, all personal information related to the record will be violated. How to protect data privacy and security in edge-assisted computing of the Internet of Things is an important issue. At present, SVM based privacy preserving research is mainly based on homomorphic encryption methods, but there are still the following defects:

First, once the classifier is handed over to the cloud server for processing, the copyright of the user classifier model will be damaged, and encryption is required for processing. Raymond et al. [8] proposed a privacy preserving classifier evaluation protocol for both parties. Compared with existing technologies, the protocol significantly improves efficiency. Zhou et al. [9] proposed a new scheme to achieve secure outsourced storage and k-NN query in the cloud, protect the privacy of users from the cloud, and users do not need to query online. However, the models of the above two schemes are stored in the server, and the adversary attacks the server and steals the models.

Second, users in the existing scheme use the same public key when uploading encrypted query data, and the security assumption is that the server cannot collude with any user. Once they collude with each other, cloud server can decrypt and obtain all users' query data and models. Recently, Meng et al. [10] proposed a scheme to support multiple users through privacy protection, allowing users to encrypt image features with the same key, thus realizing efficient image retrieval of images collected from multiple sources. However, since the data source has the same public key and private key, as long as one party obtains the encrypted data of the other party, it can decrypt the data.

Third, in the existing scheme, users and model owners must be online at the same time, interact with the cloud server, and participate in the classification stage in the whole process. Although users only need to bear a small part of computing and storage costs, they need to participate in the whole classification stage. However, in practical applications, users may not be able to maintain "online" all the time due to network conditions and other reasons, especially when there are multiple users, which brings inconvenience to data analysis.

In this paper, we further proposed a SVM classifier model over encrypted data (SVMCM-ED) based on multi-key FHE in IoT. We highlight our contributions below.

- Firstly, we propose a SVM classifier model over encrypted data in IoT to obtain multiple user classification results. The classification process is completed by the edge node and the cloud server. After sending the query data, the user can go offline. In this process, not only can data leakage be prevented, but also communication overhead of cloud server can be reduced.
- Secondly, we design two secure computing protocols based on multi-key FHE to prevent collusion between users, and provide security proof. This protocol allows encrypted calculation under different public keys and protects the data and models. Since each users has a different public and private key, no information will be leaked to the cloud server and edge node even if any users colludes with it.

## 2 Related Work

At present, researchers have done a lot of research work on privacy protection issues [11–14]. In the research of privacy protection data mining based on cryptography, there are the following problems: 1) The privacy and security of training data. 2) User query data and classification results are private data of users, including sensitive data, and their security and privacy should also be protected. Since user query data and classification results are the input and output of the classification stage, this is the privacy of the classification stage. protection issues. 3) The data is transmitted in plain text, and there is no guarantee that the data will not be stolen during the transmission process, resulting in privacy leakage. 4) For encrypted data, although fully homomorphic encryption can satisfy arbitrary operations, it is inefficient, and does not support comparison and maximal value operations.

In other works, edge computing is proposed to aid the training of industrial clouds. Edge node is composed of network devices that can be deployed anywhere over a network connection. Edge nodes have certain computing, storage and autonomous capabilities, which can reduce the data processing load on resource-constrained Internet of Things devices [15]. Since the research usually assumed that the edge node and industrial cloud are in a semi-honest model [7], the edge node will honestly implement the protocols and steal their private information by interaction with other participants, which will bring some security problems. Usually adopt data perturbation, fully homomorphic encryption, secure multi-party computation to protect privacy of edge computing outsourcing, among which fully homomorphic encryption are more common. Scholars at home and abroad pay more attention to how to perform machine learning over lightweight outsourced computing model under multiple users. Zhou et al.

[6] proposed a lightweight secure multi-key outsourced computing scheme under the cooperative but non-collusive double servers, and studied an efficient privacy preserving integer comparison protocol for wireless Internet of vehicles on this basis. Both user's location and interest privacy are effectively protected, which can resist the collusive between roadside units and semi-honest servers. However, users need to participate in the training process and bear part of the cost. The training process of the model we designed is jointly completed by the edge node and the industrial cloud. After smart device sends the training data, it can go offline without any communication overhead. The work [16] designed a privacy protection edge intelligent data aggregation solution to ensure data confidentiality, integrity, and real-timeness. The work [17] proposes a secure and smart communication solution for pervasive edge computing in infrastructure supporting IIoT. In the proposed scheme, the IIoT device detects the counterfeit identity of the adversary and shares it with the edge server to prevent the upstream transmission of its malicious data. The work [18] proposes a device-oriented anonymous privacy protection scheme with identity verification for data aggregation applications in the fog-enhanced IoT system. It also supports multiple permissions to locally manage smart devices and fog nodes.

At present, there is no Multi-key FHE system based on svm classification algorithm.

## 3 Our Proposed Design

The system architecture is shown in Fig. 1. There are four entities: User ($U_i$), Data Analysis ($DA$), Cloud Server ($CS$), and edge node ($EN$). User (e.g., client) has a large amount of user privacy data, which will be disclosed when it is handed over to $EN$ and $CS$ for computing, so it needs to be encrypted before uploading. In the medical environment, users (e.g., residents) submit personal information (e.g., weight, height, blood pressure, medical history and other personal information) to the $CS$ to obtain their health condition. The model is stored in the $DA$. The model is proprietary intellectual property and cannot be disclosed to unauthorized entities. Therefore, it is necessary to send the encrypted model to the $ED$ and $CS$. After the classification results are calculated, the user downloads it to the local for decryption to obtain the classification results (e.g., condition). The outstanding advantage of our system is that it allows $U_i$ and $DA$ to go offline completely after providing data, while the $ED$ share the computing cost of the $CS$.

## 4 Our Proposed Design

### 4.1 Overview

There are n users $U_1, U_2, ..., U_n$, and each of them has a piece of private breast cancer disease data, respectively $m_1, m_2, ..., m_n$. $DA$ has a trained SVM classifier model for breast cancer, and the classifier decision function is $f(x) =$

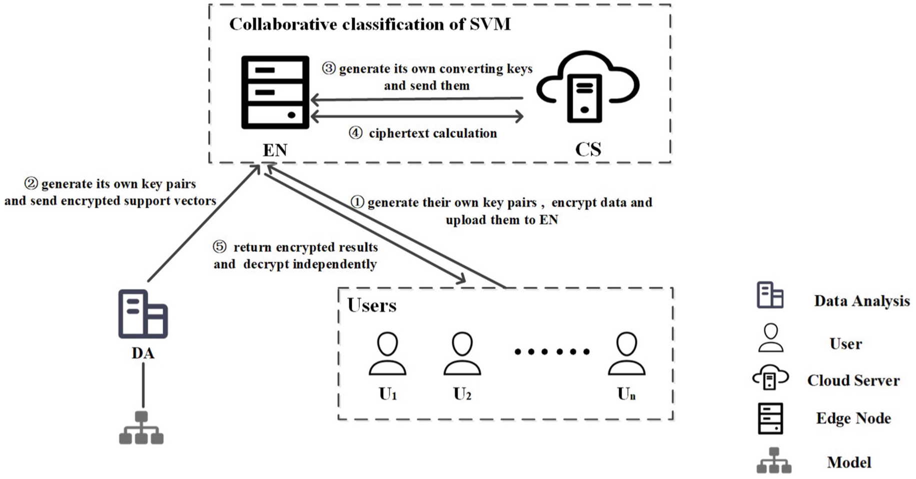

Collaborative classification of SVM
③ generate its own converting keys and send them
④ ciphertext calculation
EN
CS
② generate its own key pairs and send encrypted support vectors
① generate their own key pairs , encrypt data and upload them to EN
⑤ return encrypted results and decrypt independently
DA
Users
U1
U2
Un
Data Analysis
User
Cloud Server
Edge Node
Model

**Fig. 1.** The system architecture.

$sign(\sum_{j=1}^{t} \alpha_j^* y_j K(g_j, x) + b^*)$. $U$ wants to use $CS$ to classify data through the SVM classifier model. $U$ and $DA$ must ensure that any unauthorized third party can obtain private data information in this process.

The privacy-preserving SVM classifier requirements in this scenario can be summarized as follows:

(1) The privacy data $m_1, m_2, ..., m_n$ of any $U$ cannot be disclosed to other users, $CS$, and $DA$;
(2) The support vector $g_1, g_2, ..., g_t$ contained in the SVM classifier model owned by the $DA$ cannot be disclosed to $U_1, U_2, ..., U_n$ and $CS$;
(3) The final classification results cannot be disclosed to $CS$ and $DA$;
(4) Users bear as little computing and communication load as possible and do not need to participate in the whole classification process, that is, they do not need to be online all the time.

To meet the above requirements, this paper proposes a SVM classifier model over encrypted data (SVMCM-ED) based on multi-key homomorphic encryption to achieve the binary classification of encrypted data. The SVMCM-ED scheme uses CZW multi-key FHE algorithm to protect data privacy. Each user independently generates a public key to encrypt data and then uploads it. The $DA$ independently generates a public key to encrypt data related to the classification model and then uploads it. Introduce an edge node $ED$ that does not conspire with $CS$. The SVM classification process is jointly completed by $CS$ and $ED$ through interaction, without the participation of other parties. After the classification is completed, each user downloads $CS$'s ciphertext results and decrypts them locally with the corresponding private key.

### 4.2 Communication Protocol

This paper mainly considers privacy-preserving SVM algorithm of classification process, namely privacy-preserving SVM classifier. Based on different kernel functions, different SVM classifiers can be constructed. In order to achieve data privacy protection, we in this paper use CZW multi-key FHE algorithm, and construct secure protocols to implement classification decision functions in different types of privacy-preserving SVM classifiers, which are secure inner product classification decision protocol and secure polynomial classification decision protocol.

**Secure Inner Product Classification Decision Protocol.** Given the unlabeled data object $m_i = (m_1, m_2, ..., m_d)$, the support vector $g_1, g_2, ..., g_t$ included in the training model, the corresponding class label $y_1, y_2, ..., y_t$ and lagrange multiplier ${\alpha_1}^*, {\alpha_2}^*, ..., {\alpha_t}^*$ of the support vector, and the hyperplane displacement $b^*$. The inner product between the data object $m_i = (m_1, m_2, ..., m_d)$ and the support vector $g_1, g_2, ..., g_t$ can be expressed as:

$$K(g_j, m_i) = \sum_{z=1}^{d} g_{jz} m_{iz} \tag{1}$$

Let $h_i = \sum\limits_{j=1}^{t} \alpha_z^* y_j K(g_j, m_i) + b^*$ represent the decision value of the classification decision function. If $h(g, m_i) = 1$, then $y_i = +1$; If $h(g, m_i) = 0$, then $y_i = -1$. The security inner product classification decision protocol aims to calculate the ciphertext decision value without disclosing the plaintext.

Let $C_i = (c_1, c_2, ..., c_d)$ represent the ciphertext encrypted by $m_i$ under public key $pk_{l,i}$, and the corresponding decryption private key is $sk_{l,i}$; Let $SV_1, SV_2, ..., SV_t$ represent the ciphertext encrypted by the support vector under public key $pk_{l,j}$, and the corresponding decryption private key is $sk_{l,j}$. $A$ and $B$ are used to represent the participants of the protocol, where $A$ owns the ciphertext $C_i$,$SV_1, SV_2, ..., SV_t$ and model data $y_1, y_2, ..., y_t$,${\alpha_1}^*, {\alpha_2}^*, ..., {\alpha_t}^*$ and $b^*$ and $B$ owns the corresponding private key $sk_{l,i}$ and $sk_{l,j}$ to decrypt.

The secure inner product classification decision protocol is shown below.

**Secure Polynomial Classification Decision Protocol.** Given the unlabeled data object $m_i = (m_1, m_2, ..., m_d)$, the support vector $g_1, g_2, ..., g_t$ included in the training model, the corresponding class label $y_1, y_2, ..., y_t$ and lagrange multiplier ${\alpha_1}^*, {\alpha_2}^*, ..., {\alpha_t}^*$ of the support vector, and the hyperplane displacement $b^*$. The polynomial kernel function between the data object $m_i = (m_1, m_2, ..., m_d)$ and the support vector $g_1, g_2, ..., g_t$ can be expressed as:

$$K(g_j, m_i) = (\sum_{z=1}^{d} g_{jz} m_{iz} + 1)^u \tag{2}$$

Protocol 2 is a description of the dot product protocol.

**Protocol 1.** Secure inner product classification decision protocol

```
Input A: $C_i$,$SV_1, SV_2, ..., SV_t$,$y_1, y_2, ..., y_t$,$\alpha_1{}^*, \alpha_2{}^*, ..., \alpha_t{}^*$ and $b^*$;
Input B: private key $sk_{l,i}$ and $sk_{l,j}$;
Output A: classification result $C\left(h_i\right)$
A:
if $pk_{l,i} \neq pk_{l,j}$ then
    expand $C_i$,$SV_1, SV_2, ..., SV_t$ to $\overline{C_i}$, $\overline{SV_1}, \overline{SV_2}, ..., \overline{SV_t}$, and the corresponding key
    is $sk_{l,\overline{U}} = (sk_{l,i}|sk_{l,j})$, where $\overline{U} = \{i, j\}$
    compute $s_i = \sum_{j=1}^{t} \alpha_j^* y_j s_{j,i}$ , and the corresponding key is $sk'_{l,\overline{U}} = (sk_{l,\overline{U}} \otimes sk_{l,\overline{U}})$
else
    compute $s_i = \sum_{j=1}^{t} \alpha_j^* y_j s_{j,i}$ , where the corresponding key is $sk'_{l,\overline{U}} = (sk_{l,i} \otimes sk_{l,j})$
end if
refreshing $s_i$ is $s_i{}'$, and the corresponding key of $s_i{}'$ is $sk_{l-1,\overline{U}}$
select random number $r$, and encrypt $r, b^*$ under public key $pk_{l-1,\overline{U}}$ to get
$\overline{C}(r), \overline{C}(b^*)$
compute $\widetilde{h_i} = s_i' + \overline{C}(b^*) + \overline{C}(r)$ and send $\widetilde{h_i}$ to $B$
B:
decrypt $\widetilde{h_i}$ to $\widetilde{h_i}'$, and encrypt $C(\widetilde{h_i}')$ to $A$
A:
encrypt $r$ under public key $pk_{l,i}$ to $C(r)$ and calculate $C(h_i) = C(\widetilde{h_i}') - C(r)$
```

### 4.3 Classification Process

SVMCM-ED includes three stages, namely, secure data encryption and upload stage, security data classification stage and secure decryption stage. The specific description is as follows:

**Secure Data Encryption and Upload Stage.** In this stage, each user $U_i(i \in [n])$ must independently generate its own key pair $\{pk_{l,i}, sk_{l,i}\}_{l=\{L,...,1,0\}}$, encrypt the private data $m_i$ with the public key $pk_{L,i}$, and generate the converting key $\left\{\tau_{sk_{L.j} \to sk_{L,S}}\right\}$.

$$\left\{\tau_{sk_{L.i} \to sk_{L,S}}\right\} \leftarrow MKFHE.SwitchKeyGen(pp, sk_{L,j}, pk_{L,S})a \tag{3}$$

The data analysis $(DA)$ independently generates its own key pair $\{pk_{l,DA}, sk_{l,DA}\}$, and encrypts $g_1, g_2, ..., g_t$ under the public key $pk_{l,DA}$ to obtain the ciphertext $SV_1, SV_2, ..., SV_t$, and generates the converting key $\left\{\tau_{sk_{L.DA} \to sk_{L,S}}\right\}$.

$$\left\{\tau_{sk_{L.DA} \to sk_{L,S}}\right\} \leftarrow MKFHE.SwitchKeyGen(pp, sk_{L,DA}, pk_{L,S}) \tag{4}$$

$CS$ generates its own key pair $\{pk_{l,s}, sk_{l,s}\}_{l=\{L,...,1,0\}}$ independently and the converting key $\left\{\tau_{sk_{l.S} \to sk_{l-1,S}}\right\}$ for refresh the ciphertext.

In this stage, each data owner $U_i(i \in [n])$ independently uploads its ciphertext data $C_i$, public key $pk_{L,i}$ and the converting key $\left\{\tau_{sk_{L.j} \to sk_{L,S}}\right\}$ to the edge node

**Protocol 2.** Secure polynomial classification decision protocol

```
Input A: $C_i$,$SV_1, SV_2, ..., SV_t$,$y_1, y_2, ..., y_t$,$\alpha_1{}^*, \alpha_2{}^*, ..., \alpha_t{}^*$ and $b^*$;
Input B: private key $sk_{l,i}$ and $sk_{l,j}$;
Output A: classification result $C(h_i)$
A:
if $pk_{l,i} \neq pk_{l,j}$ then
    expand $C_i$,$SV_1, SV_2, ..., SV_t$ to $\overline{C_i}$, $\overline{SV_1}, \overline{SV_2}, ..., \overline{SV_t}$, and the corresponding key is $sk_{l,\overline{U}} = (sk_{l,i}|sk_{l,j})$, where $\overline{U} = \{i, j\}$
    compute $s_i = \sum_{j=1}^{t} \alpha_j^* y_j s_{j,i}$ , and the corresponding key is $sk'_{l,\overline{U}} = (sk_{l,\overline{U}} \otimes sk_{l,\overline{U}})$
else
    compute $s_i = \sum_{j=1}^{t} \alpha_j^* y_j s_{j,i}$ , where the corresponding key is $sk'_{l,\overline{U}} = (sk_{l,i} \otimes sk_{l,j})$
end if
refreshing $s_i$ is $s_i{}'$, and the corresponding key of $s_i{}'$ is $sk_{l-1,\overline{U}}$
select random number $r$, and encrypt $r, b^*$ under public key $pk_{l-1,\overline{U}}$ to get $\overline{C}(r), \overline{C}(b^*)$
compute $\widetilde{h_i} = \sum_{j=1}^{t} \alpha_z^* y_j p_{j,i} + \overline{C}(b^*) + \overline{C}(r)$, where the corresponding key is $sk_{l-u,\overline{U}} = (sk_{l-u,i}|sk_{l-u,j})$ and send $\widetilde{h_i}$ to $B$
B:
decrypt $\widetilde{h_i}$ to $\widetilde{h_i}'$, and encrypt $C(\widetilde{h_i}')$ to $A$
A:
encrypt $r$ under public key $pk_{l',i}$ to $C(r)$ and calculate $C(h_i) = C(\widetilde{h_i}') - C(r)$
```

$ED$ through the secure channel. The $DA$ independently uploads the kernel function $K(x, z)$, ciphertext $SV_1, SV_2, ..., SV_t$, plaintext $y_1, y_2, ..., y_t$,$\alpha_1{}^*, \alpha_2{}^*, ..., \alpha_t{}^*$, public key $pk_{l,DA}$ and converting key $\{\tau_{sk_{L.DA} \to sk_{L,S}}\}$ used in the SVM classifier to the $ED$ through a secure channel. The $CS$ uploads public key $pk_{l,S}$ and converting keys $\{\tau_{sk_{l.S} \to sk_{l-1,S}}\}$ to $ED$. At this time, $ED$ runs the key exchange sub-algorithm to convert $C_1, C_2, ..., C_n$, $SV_1, SV_2, ..., SV_t$ into $C_{S1}, C_{S2}, ..., C_{Sn}$, $SV_{S1}, SV_{S2}, ..., SV_{St}$. After conversion, the private keys corresponding to the ciphertext are $sk_{L,S}$.

**Secure Classification Decision Stage.** In this stage, $CS$ and $ED$ complete the classification decision of data object $m_i = (m_1, m_2, ..., m_d)$ through security protocols, and different kernel functions select different security protocols. $ED$ gets the ciphertext decision value $h_{S1}, h_{S2}, ..., h_{Sn}$, and converts the ciphertext to $C(h_1), C(h_2), ..., C(h_n)$.

$$C(h_i) \leftarrow MKFHE.SwitchKey(\tau_{sk_{l'},s \to sk_{l',i}}, h_{Si}) \tag{5}$$

**Secure Decryption Stage.** In this stage, $U_i$ downloads ciphertext $C(h_i)$ from $ED$ and decrypts decision value $h_i$ independently with private key $sk_{l',i}$.

$$h_i \leftarrow MKFHE.Dec(pp, ct = (h, i, l'), sk_{l',i}) \tag{6}$$

That is, the classification label of $m_i$ is $y_i = sign(h_i)$. If $h_i = 1$, then $y_i = 1$; If $h_i = 0$, then $y_i = -1$.

## 5 Security Analysis

This section analyzes the security of the communication protocol and SVMCM-ED under the semi-honest model. Both $CS$ and $EN$ are semi-honest participants, they honestly follow the execution of the protocol and allow inferences from the data obtained during the execution of the protocol. Its input data is private data and can only be known by the individual.

It is divided into the following three parts to discuss the security of the classifier under the semi honest model:

(1) Secure data encryption and upload stage: At this stage, $U_1, U_2, ..., U_n$ generates ciphertext $C_i$ and converting key $\{\tau_{sk_{L.j} \to sk_{L,S}}\}$ independently and stores them on $ED$. $DA$ generates ciphertext $SV_1, SV_2, ..., SV_t$ and converting key $\{\tau_{sk_{L.DA} \to sk_{L,S}}\}$ independently and stores them on $ED$. The independent generation key $\{\tau_{sk_{l.S} \to sk_{l-1,S}}\}$ of $CS$ is stored on $ED$. The $ED$ then converts $C_1, C_2, ..., C_n$, $SV_1, SV_2, ..., SV_t$ into $C_{S1}, C_{S2}, ..., C_{Sn}$, $SV_{S1}, SV_{S2}, ..., SV_{St}$.

(2) Secure classification decision stage: At this stage, $ED$and $CS$ calculate the decision value of privacy data through different secure classification decision protocols according to different kernel functions $K_(x, z)$.

In the semi honest model, protocol participants $A$ and $B$ perform protocol operations honestly, but $A$ is curious about data objects $m_i$, support vectors $g_1, g_2, ..., g_t$, and decision value $h_i$, and is curious about decision value $h_i$.

The secure inner product classification decision protocol only involves the addition and multiplication homomorphic operations between ciphertext. Similarly, the power function calculation in the secure polynomial classification decision protocol is converted into multiple ciphertext multiplication calculations, so the protocol only involves the addition and multiplication homomorphic operations between ciphertext. For $A$, without the private key $sk_{l,i}$ and $sk_{l,j}$, according to the semantic security of the CZW scheme, $A$ cannot infer any information of the corresponding plaintext from the ciphertext $C_i$, $SV_1, SV_2, ..., SV_t$, $C(h_i)$. Since $A$ introduces random number $r$, $B$ cannot recover the decision value $h_i$ from $\widetilde{h_i}$.

In conclusion, the secure inner product classification decision protocol and the secure polynomial classification decision protocol mentioned in this section are safe under the semi-honest model. Therefore, $ED$ cannot obtain any data information in the classification decision. Although $CS$ has a private key that can be decrypted, because $ED$ is disturbed by calculation or random number before sending the ciphertext, $CS$ cannot obtain any private data information during the classification decision process.

(3) Secure decryption stage: At this stage, $U_1, U_2, ..., U_n$ decrypts $h_1, h_2, ..., h_n$ independently and judges the classification label $y_1, y_2, ..., y_n$ of $m_1, m_2, ..., m_n$. According to the semantic security of CZW multi-key FHE algorithm, any third-party without the corresponding decryption private key $sk_{l,1}$, $sk_{l,2}, \ldots$, $sk_{l,n}$ can obtain the information of $h_1, h_2, ..., h_n$ and $y_1, y_2, ..., y_n$.

To sum up, the proposed SVMCM-ED scheme is safe under the semi honest model.

## 6 Experiments

Our protocols are implemented in C++ on Ubuntu 18.04.2 64-bit version, Inter(R) Core(TM) i7-9700M CPU (3.00 GHz) 32 GB RAM. The server with the same configuration were used to simulate the edge node ($EN$) and the cloud server ($CS$) to collaborative implement the SVM model together in order to reduce the burden on the cloud server.

Four FCPS [18] standard datasets are used, namely: Hepta, Lsun, Tetra, and Wingnut public datasets, which are specially used for classification analysis. To facilitate the experiment, the range of security parameters of the encryption algorithm selected in this paper is $5 < \kappa < 60$. Wingnut dataset is used.

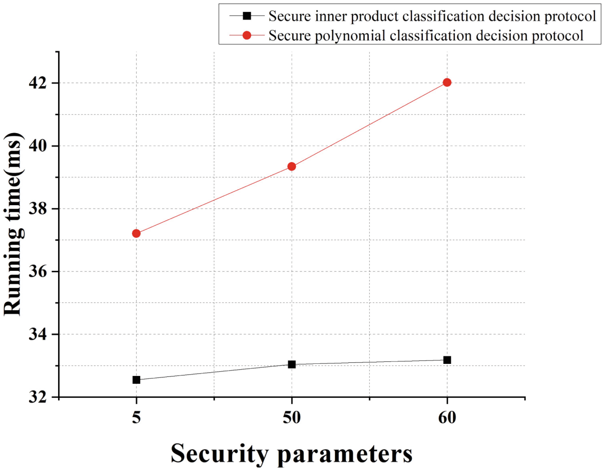


**Fig. 2.** Running time of two communication protocols.

The efficiency of each protocol is shown below. Figure 2 respectively shows the running time of the secure inner product classification decision protocol and secure polynomial classification decision protocol when the security parameter $\kappa = 5, 50, 60$.

The following experiments were conducted under the condition of $\kappa = 50$. As shown in Table 1, the feature descriptions of the four datasets are given. The effect of iteration number $T$ on the classification effect of plaintext SVM and SVMCM-ED is discussed through experiments and analysis.

**Table 1.** Accuracy comparison of SVM and SVMCM-ED

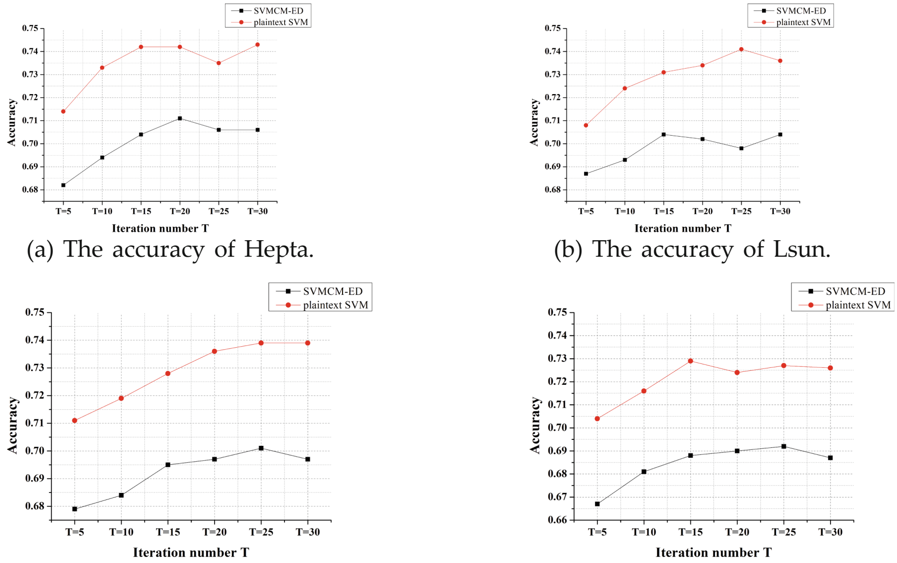
| Data | Model | Accuracy | | | | | |
|---|---|---|---|---|---|---|---|
| | | T = 5 | T = 10 | T = 15 | T = 20 | T = 25 | T = 30 |
| Hepta | plaintext SVM | 0.714 | 0.733 | 0.742 | 0.742 | 0.735 | 0.743 |
| | SVMCM-ED | 0.682 | 0.694 | 0.704 | 0.711 | 0.706 | 0.706 |
| Lsun | plaintext SVM | 0.708 | 0.724 | 0.731 | 0.734 | 0.741 | 0.736 |
| | SVMCM-ED | 0.687 | 0.693 | 0.706 | 0.702 | 0.698 | 0.704 |
| Tetra | plaintext SVM | 0.711 | 0.719 | 0.728 | 0.736 | 0.739 | 0.739 |
| | SVMCM-ED | 0.679 | 0.684 | 0.695 | 0.697 | 0.701 | 0.697 |
| Wingnut | plaintext SVM | 0.704 | 0.716 | 0.729 | 0.724 | 0.727 | 0.726 |
| | SVMCM-ED | 0.667 | 0.681 | 0.688 | 0.690 | 0.692 | 0.687 |

The accuracy is used to measure the classification effect. Table 1 and Fig. 3 show the comparison of the accuracy of plaintext SVM and SVMCM-ED with different values of $T$ selected. When $T = 5$–15, the accuracy rate increases rapidly, which means that the classification performance is significantly improved; while when $T = 15$–30, the accuracy rate does not change much. After 15 rounds, no further iterations are required.

From the data analysis in Fig. 3, it can be seen that the accuracy of SVMCM-ED is 2.9%–5.8% lower than that of the plaintext SVM algorithm, and the accuracy of SVMCM-ED is above 0.6, which is within the acceptable range. Therefore, the model constructed in this paper can be used. The SVM algorithm is completed in this paper, and the classification effect is guaranteed.

**Table 2.** Time cost comparison of plaintext classification and SVMCM-ED

| The running time (ms) | The total number of data | | | | |
|---|---|---|---|---|---|
| | 50 | 100 | 200 | 300 | 400 |
| plaintext SVM | 84 | 91 | 176 | 271 | 352 |
| SVMCM-ED egde node | 3343.81 | 6253.16 | 12207.36 | 18847.42 | 23497.92 |
| SVMCM-ED cloud server | 2788.11 | 5218.49 | 10284.21 | 15745.26 | 20935.43 |



(a) The accuracy of Hepta.

(b) The accuracy of Lsun.

(c) The accuracy of Tetra.

(d) The accuracy of Wingnut.

**Fig. 3.** Accuracy comparison of SVM and SVMCM-ED.

Using the Lsun data set, $n$ represents the total number of data, taking $n = 50$, 100, 200, 300, 400, and conduct experiments respectively. Table 2 shows the classification of the plaintext SVM algorithm and the running time of the SVMCM-ED egde node and cloud server when $n$ changes. The data in the table shows that its running time increases linearly with $n$, and the running time of ciphertext classification is between 1 s and 24 s, which is within the acceptable range. The egde node bears the main overhead, reducing the burden on the cloud server.

## 7 Conclusion

In this paper, a SVM classifier model over encrypted data (SVMCM-ED) is implemented instead of using the SVM classifier model directly. It allows data analysis to provide their encrypted model so that edge nodes can cooperate with cloud server and outsource SVM classifier services. We give the entity structure of the model and design the communication protocols, including the secure inner product classification decision protocol and secure polynomial classification decision protocol. More importantly, the module sequence combination is constructed based on the above protocol. Then, we give the security analysis. The experiments show that SVMCM-ED can realize the SVM classification over encypted data, and the edge nodes can share the burden of cloud servers.

## References

1. Li, T., Huang, Z., Li, P., Liu, Z., Jia, C.: Outsourced privacy-preserving classification service over encrypted data. J. Netw. Comput. Appl. **106**(15), 100–110 (2018)
2. Xie, B., Xiang, T., Liao, X.F.: Access-oblivious and privacy-preserving k nearest neighbors classification in dual clouds. Comput. Commun. **187**, 12–23 (2022)
3. Ren, S., Kim, J., Cho, W.S., Soeng, S., Kong, S., Lee, K.H.: Big data platform for intelligence industrial IoT sensor monitoring system based on edge computing and AI. In: Proceedings of the ICAIIC 2021 (2021)
4. Gao, W.F., Zhao, Z.W., Yu, Z.X., Min, G.Y., Yang, M.H., Huang, W.J.: Edge computing based channel allocation for deadline-driven IoT networks. IEEE Trans. Ind. Inf. **16**(10), 6693–6702 (2020)
5. Zhou, J., Cao, Z., Qin, Z., Dong, X.L., Ren, K.: LPPA: lightweight privacy-preserving authentication from efficient multi-key secure outsourced computation for location-based services in VANETs. IEEE Trans. Inf. Forensics Secur. **15**(10), 420–434 (2019)
6. So, J., Guler, B., Avestimehr, S.: CodedPrivateML: a fast and privacy-preserving framework for distributed machine learning. IEEE J. Sel. Areas Inf. Theory **2**(1), 441–451 (2021)
7. Tai, R.K.H., Ma, J.P.K., Zhao, Y., Chow, S.S.M.: Privacy-preserving decision trees evaluation via linear functions. In: Foley, S.N., Gollmann, D., Snekkenes, E. (eds.) ESORICS 2017. LNCS, vol. 10493, pp. 494–512. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66399-9_27
8. Lu, Z., Zhu, Y., Castiglione, A.: Efficient k-NN query over encrypted data in cloud with limited key-disclosure and offline data owner. Comput. Secur. **69**, 84–96 (2017)
9. Shen, M., Cheng, G., Zhu, L., Du, X., Hu, J.: Content-based multi-source encrypted image retrieval in clouds with privacy preservation. Future Generation Computer Systems
10. Bost, R., Pop, R., Tu, S.: Machine learning classification over encrypted data. In: Network & Distributed System Security Symposium (2014)
11. Zheng, Y., Duan, H., Wang, C., Wang, R.: Securely and efficiently outsourcing decision tree inference. IEEE Trans. Dependable Secure Comput. 1 (2020)
12. Xu, R., Joshi, K., Li, C.: NN-EMD: efficiently training neural networks using encrypted multi-sourced datasets. IEEE Trans. Dependable Secure Comput. 1 (2021)
13. Crawford, J.L.H., Gentry, C., Halevi, S., Platt, D.: Doing Real Work with FHE: The Case of Logistic Regression (2018)
14. Chamikara, M., Bertok, P., Khalil, I., Liu, D., Camtepe, S.: Privacy preserving distributed machine learning with federated learning. Comput. Commun. **171**(4), 112–125 (2021)
15. Xiong, J., et al.: A personalized privacy protection framework for mobile crowdsensing in IIoT. IEEE Trans. Ind. Inform. **16**(/6), 4231–4241 (2020)
16. Khan, F., Jan, M.A., Rehman, A., Mastorakis, S., Alazab, M., Watters, P.: A secured and intelligent communication scheme for IIoT-enabled pervasive edge computing. IEEE Trans. Ind. Inf. **17**(7), 5128–5137 (2021)
17. Guan, Z.T., et al.: APPA: an anonymous and privacy preserving data aggregation scheme for fog-enhanced IoT. J. Netw. Comput. Appl. (2019)
18. FCPS[EB/OL]. http://uni-marburg.de/fb12/datenbionik/data?languagesvnc=1/