Hindawi
Wireless Communications and Mobile Computing
Volume 2024, Article ID 8138644, 15 pages
https://doi.org/10.1155/2024/8138644

Research Article

Federated Medical Learning Framework Based on Blockchain and Homomorphic Encryption

Xiaohui Yang and Chongbo Xing

School of Cyber Security and Computer, Hebei University, Baoding 071000, China

Correspondence should be addressed to Chongbo Xing; 1660304154@qq.com

Received 28 November 2022; Revised 8 November 2023; Accepted 13 December 2023; Published 5 January 2024

Academic Editor: Yan Huang

Federated learning-based medical data privacy sharing can promote the development of medical industry intelligence, but limited by its own security and privacy deficiencies, federated learning still suffers from a single point of failure and privacy leakage of intermediate parameters. To address these problems, this paper proposes a privacy protection framework for medical data based on blockchain and cross-silo federated learning, using cross-silo federated learning to establish a collaborative training platform for multiple medical institutions to enhance the privacy of medical data, introducing blockchain and smart contracts to realize decentralized federated learning to enhance trust between distrustful medical institutions and solve the problem of a single point of failure. In addition, a secure aggregation scheme is designed using threshold homomorphic encryption to prevent the privacy leakage problem during parameter transmission. The experimental and analytical results show that the accuracy of this paper's scheme is consistent with the original federated learning scheme, effectively deals with the problems of single-point failure and inference attacks of federated learning, improves system robustness, and is suitable for medical scenarios with more stringent requirements on security and accuracy.

1. Introduction

With the booming development of emerging intelligent technologies, the healthcare industry is rapidly developing and gradually entering the era of intelligence. At this stage, medical data with explosive growth characteristics can help healthcare professionals use AI applications to better diagnose diseases, but the data of individual medical institutions cannot meet the needs of AI applications, and at the same time, because of the low level of trust among medical institutions and the large amount of patient privacy information contained in medical data, there is data isolation among medical institutions, and the noncirculation of medical data makes the collection of medical data very difficult. In addition, legal access to data has become a worldwide trend, and several countries, including China, have proposed bills to regulate data collection, emphasizing the legality of data sharing. Therefore, reasonable and legal access to medical data has become a hot topic of current research.

Federated learning (FL) [1, 2] is a distributed collaborative learning model that does not require a centralized collection of users' private data for training but only requires users to train on local datasets and upload the trained local models to an aggregator, through which the local models are aggregated to generate a new global model. Depending on the initial setup, federated learning can be divided into cross-device federated learning and cross-silo federated learning [3]. Among them, cross-silo federated learning allows a small number of organizations to collaborate with institutions to train machine learning models. The emergence of federated learning has opened up new horizons in the field of artificial intelligence, and cross-silo federated learning helps to break the data isolation among medical institutions, better interact with medical privacy data, and promote the development of intelligence in the medical industry.

However, there are a number of challenges in applying federated learning technologies in the healthcare field.

The first is the trust issue. Since the participants of federated learning come from different medical institutions and lack trust among themselves, it is especially important to establish a secure cooperation mechanism in the absence of trust. Second is the single point of failure problem. The federated learning framework uses a fixed central server as an aggregator, which can cause this training to fail due to possible security attacks on the device and damage to the physical equipment. Finally, there is the inference attack, where the central server can analyze and derive the user's raw data from the local model updates provided by the trainer during the training process, which can expose the patient's privacy.

Blockchain is transparent, tamper-evident, and auditable [4], which can provide a trusted platform for federated learning and solve the single point of failure problem. Blockchain is essentially a distributed ledger that uses cryptographic techniques to secure data on the chain while building a safe and secure data-sharing platform for multiple untrustworthy participants through its use of consensus mechanisms, smart contracts [5], and other technologies. However, the data recorded in the blockchain is open to the whole network, and curious participants can easily access the required data from the blockchain, increasing the risk of data leakage.

Homomorphic encryption is a cryptographic technique that enables computation on encrypted data and is widely used in scenarios where data privacy computation needs exist, such as blockchain and federated learning. However, the application of original homomorphic encryption in federated learning relies on the honesty of the key holder, and the node with the private key of homomorphic encryption can easily decrypt the encrypted data, which can compromise the data privacy of the training participants.

To solve the problem of medical privacy data sharing among medical institutions, this paper introduces blockchain into cross-silo federated learning to provide a trusted platform for medical institutions that do not trust each other and uses smart contracts to regulate the federated learning process and select appropriate aggregators for secure aggregation of local models to achieve decentralized federated learning. In addition, a threshold Paillier cryptosystem [6] is used to address possible inference attacks during the training process and to ensure data privacy during parameter delivery.

The contributions of this paper are summarized as follows:

(1) A privacy protection framework for medical data based on blockchain and federated learning is proposed, which not only provides a secure and trustworthy data-sharing platform for medical data but also makes it tamper-proof and auditable.

(2) A secure aggregation scheme based on threshold homomorphic encryption is designed to ensure the secure aggregation of model parameters and prevent local model parameters from leaking local data privacy during the transmission process.

(3) A smart contract for secure upload and aggregation node selection of local model parameters is designed to solve the single point of failure problem in the federated learning process through the dual guarantee of blockchain and smart contract. And the IPFS file system [7] is used to reduce the storage pressure of the blockchain.

(4) The framework proposed in this paper is tested and evaluated to demonstrate that it improves the privacy and security of medical data sharing while maintaining accuracy with the traditional federated learning scheme.

2. Related Work

2.1. Federated Learning in the Medical Field. Federated learning has been widely used to break down "data silos" among medical institutions and share private patient data for medical research. Brisimi et al. [8] developed a federated learning model for predicting future hospitalizations of cardiac patients without interacting with the user with raw data. Silva et al. [9] proposed a federated learning framework for securely accessing any biomedical data without sharing personal information. Sheller et al. [10] applied federated learning to healthcare to facilitate collaboration among multiple healthcare providers while protecting patients' medical data. Zhang et al. [11] designed a dynamic fusion-based federated learning system architecture to analyze COVID-19 medical diagnostic images. Rieke et al. [12] analyzed how federated learning can provide solutions for the future of digital health and highlight factors to consider in practical applications. Darzi et al. [13] investigated federated learning for adversarial attacks in the field of medical image analysis and charted the future of federated learning. All of these works use federated learning to improve the security of medical data sharing but do not consider the privacy and security issues that exist with federated learning technology itself, and the security and privacy of patient data remain a threat.

2.2. Privacy Protection for Federated Learning. While the original data are kept local to the participants during the federated learning process, the local model parameters passed during the training process may still reveal patient privacy. Melis et al. [14] demonstrated that sensitive information in federated learning local model updates can be obtained through inference attacks. Hitaj et al. [15] devised an attack by using a generative adversarial network (GAN) to obtain sensitive information from local model parameters that have been used for differential privacy. An adversarial network (GAN) can obtain sensitive information from local model parameters where differential privacy has been used.

The main existing privacy-preserving methods for federated learning are Homomorphic Encryption (HE) and differential privacy (DP). Shokri et al. [16] proposed that using differential privacy in deep learning model parameters can improve its privacy but will reduce the accuracy of the model. Wu et al. [17] proposed a federated learning scheme based on differential privacy with an adaptive gradient descent strategy to improve efficiency in multiparty computation scenarios while enhancing the privacy of federated learning. Wang et al. [18] proposed a noninteractive federated learning framework, which improves federated learning privacy, but its improved Paillier homomorphic encryption scheme leads to the necessity of keeping all nodes up and running during federated learning without having down or offline nodes, which will reduce federated learning robustness. Park et al. [19] considered single-key homomorphic encryption used in federated learning with low security and designed a privacy-preserving federated learning algorithm that aggregates the local model parameters of different HE key encryptions but increases the resource consumption of homomorphic encryption. Tawose [20] focused on the security of incentives in federated learning systems and proposed a homomorphic encryption algorithm called RHE. Ma et al. [21] proposed a new privacy-preserving federated learning scheme that enhances the federated learning framework privacy by using the xMK-CKKS multikey homomorphic encryption protocol. Overall, using differential privacy to enhance federated learning privacy requires adding noise to the model or data, which inevitably affects the final usability of the model and has insufficient privacy. In contrast, using homomorphic encryption in federated learning does not affect the final usability of the model, but it increases resource consumption, and there is a security problem with single-key homomorphic encryption applied to federated learning, where an aggregator with a private key can decrypt all local model parameters encrypted using the corresponding public key by the private key.

2.3. Blockchain-Based Federated Learning. In response to the single point of failure, data security, and node multiparty trust problems of federated learning, some works introduced blockchain into federated learning. Qammar et al. [22] have systematically organized the current integration of blockchain in federated learning with an in-depth study of the existing security issues, traceability, reward, and punishment mechanisms. Javed et al. [23] proposed a scheme called Sharechain for the healthcare data privacy problem, which uses blockchain as an infrastructure and empowers the blockchain with federated learning and local differential privacy to enable the secure sharing of healthcare data. Rifai et al. [24] proposed a blockchain-based medical federated learning framework to achieve decentralized federated learning by aggregating local models of federated learning through smart contracts but did not focus on the current bottlenecks of blockchain and smart contracts. Awan et al. [25] proposed a blockchain-based privacy-preserving framework for federated learning, using homomorphic encryption and proxy re-encryption techniques to protect data privacy, but it did not consider the single point of failure of aggregators. Zhang et al. [26] designed a reputation evaluation mechanism for federated learning, using the quality of model parameters as an important indicator for selecting trainers and aggregators. Kim et al. [27] addressed the existence of a single point of failure problem by introducing blockchain in federated learning, but it ignored the privacy problem of federated learning itself. Zhang et al. [28] proposed a blockchain-based medical federated learning framework and protected patient data privacy by adding differential privacy noise, but the addition of differential noise would reduce the accuracy of the final global model. Majeed et al. [29] constructed a secure and reliable federated learning platform using Ethereum; however, its use of a single-key homomorphic confidentiality technique does not satisfy data privacy during data model delivery. Wang et al. [30] proposed a blockchain-based access control mechanism and a federated learning framework for genome-wide association studies, which enhances the security of medical data sharing through the mutual empowerment of blockchain and federated learning and the use of differential privacy techniques. Feng et al. [31] proposed an asynchronous federated learning framework based on blockchain to address the security and efficiency issues of federated learning frameworks, which achieves the nontamperability of the federated learning training process through blockchain and accelerates the aggregation process of global models through asynchronous learning. Issa et al. [32] investigated a blockchain-based federated learning framework for IoT security issues, showed the role of blockchain as well as smart contracts in a federated learning framework, and discussed the security issues related to integrating blockchain and federated learning in IoT. Qu et al. [33] investigated the problems of single point of failure as well as incentives in federated learning and showed that blockchain can improve the performance of federated learning from several perspectives. Chen et al. [34] proposed a data-sharing privacy preservation model based on blockchain and federated learning for addressing the privacy and integrity of user data.

In summary, most of the existing studies on medical data sharing based on federated learning ignore the privacy issues that exist in themselves and the impact of noise errors on the global model. In addition, in the joint framework of blockchain and federated learning, some studies do not consider the computational and storage limitations brought by the introduction of blockchain and the single point of failure risk brought by the fixed aggregator selection.

3. Materials and Methods

3.1. Overview

3.1.1. System Model. For federated learning in medical scenarios, this paper chooses to use both blockchain technology and homomorphic cryptography to enhance the security and privacy of the medical federated learning framework. As shown in Figure 1, this paper is divided into two phases: the initialization phase and the federated learning training phase; where the initialization phase includes the acquisition of keys and the release of smart contracts, which are mainly for the overall preprocessing of the upcoming federated learning tasks; followed by the federated learning phase, which is mainly for the federated learning training and encryption and decryption. The scheme in this paper consists of four entity roles: task publisher, care delivery organization, trusted authority, and aggregator.

Task publisher (TP): It is the initiator of the training task and also a node in the blockchain, which needs to use the IPFS file system for global model storage. The task publisher needs to publish tasks on the blockchain through smart contracts and verify the accuracy of the final model using local test datasets.

Care delivery organization (CDO): CDO is a federated learning task participant that has its own training dataset to train a local model by global model parameters obtained from the blockchain network and encrypt the local model

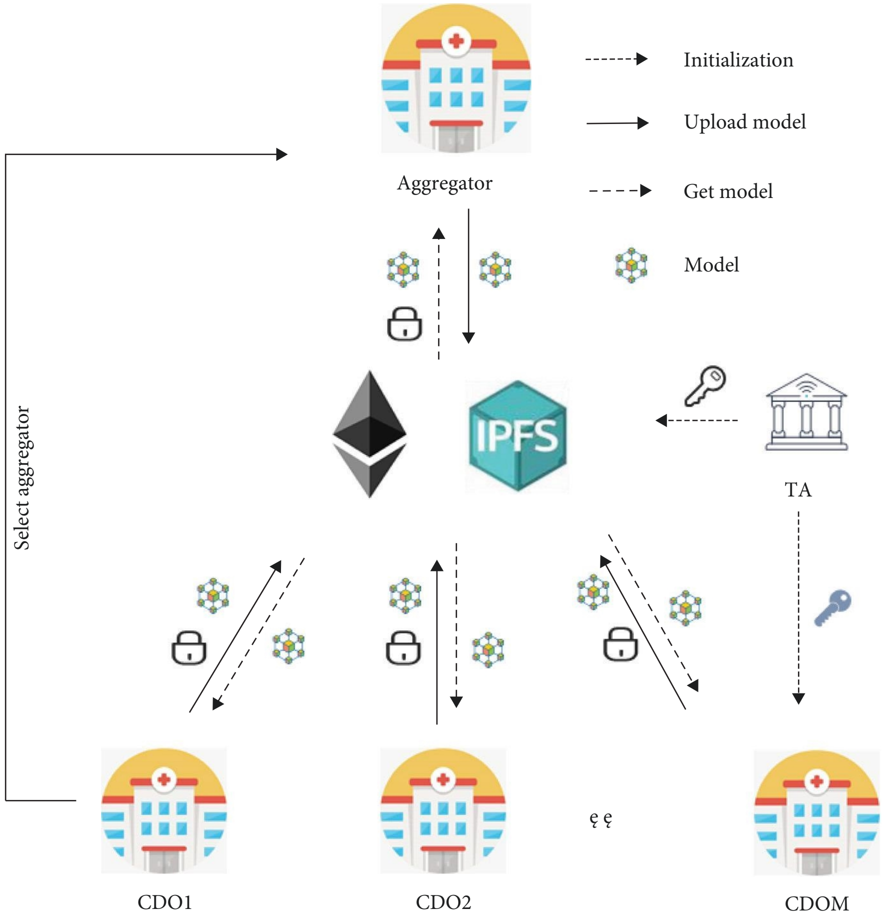


Figure 1: System framework.

parameters by obtaining PK through a smart contract. In addition, CDO can obtain SK_{CDOi} from TA and partially decrypt the global aggregated model. Each medical institution is a node in the blockchain network and uses the IPFS file system for local model storage.

Trust authority (TA): A fully trusted third party is responsible for honestly executing the cryptographic algorithm, uploading the public key PK for local model encryption via a smart contract, and distributing the homomorphic encryption key SK_{CDOi}, to the data owner, while fairly selecting the federated learning aggregation nodes by invoking the smart contract.

Aggregator: It is a CDO randomly selected by TA from the CDOs participating in this training to collect all the local models for this training round and perform aggregation. In the aggregator selection, each CDO may become an aggregator for this training round. In addition, the aggregator has to collect the partially decrypted models from the CDOs and perform the final model decryption.

3.1.2. Threat Model. Some possible security issues in the scheme are analyzed and discussed in this paper.

(1) Potential privacy leakage: For commercial interests, each CDO in the federated learning training process will be curious about the local training data of other CDOs and make efforts to obtain the data of other CDOs, even through some illegal means, e.g., inference attacks.

(2) Aggregation attack: During the process of aggregation and decryption of model parameters, an illegal attacker may attack the aggregator that is performing the aggregation operation or the CDO that is performing the decryption to disrupt the federated learning training process.

(3) Collusion attack: When T CDOs in the system exceed a threshold value for collusion, they can then decrypt the encrypted local model parameters of the uploaded blockchain by combining them into a complete private key through subkeys.

To implement the scheme in this paper, we make the following assumptions. First, since the data owners are all medical institutions, which are limited in number, and the ultimate goal of all participants is to train a high-precision medical model for use in real-world scenarios, we assume that all medical institutions involved in the training are honest but curious. Specifically, during the training process, each medical institution will do its work honestly and will not engage in poisoning behaviors that affect the training accuracy of the model. Second, it is assumed that each medical institution has sufficient computational power to make the appropriate calculations within the specified time and that the medical institutions are secure in the process of data upload. Finally, this paper does not consider the problem of a possible single point of failure of TA, which will perform the key distribution work safely and securely.

3.2. Distributed Medical Federated Learning Framework. In the medical federated learning framework, each federated learning task is published by a task publisher and trained collaboratively by M medical institutions, and each medical institution CDOi has its own dataset *Di*. At the beginning of training, TA assigns weights according to the amount of data each CDO has, and when assigning keys, since the scheme of this paper targets a small number of medical institutions, $\mathrm{SK}_{\mathrm{CDO}i}$ are assigned directly to medical institutions participating in the federated learning training. In addition, considering that the blockchain platform used in this paper is Ethereum [35], which does not allow carrying parameters for transactions initiated by ordinary accounts to ordinary accounts, it is necessary to design smart contracts to upload the model parameters to the blockchain. The specific process is shown in Figure 2.

3.2.1. Program Process. Step 1: Registration and smart contract deployment. When new medical institutions and task publishers join the network, they need to send a registration request to TA, which includes their address, the type, and amount of data they have. TA deploys smart contracts, which mainly include aggregation node selection, etc. The smart contract is shown in Algorithm 1.

Step 2: Task publishing. The task publisher first uploads the initial global model parameters and other related information to IPFS and records the hash value returned by IPFS into the blockchain through a smart contract. At the same time, the TA randomly selects one of the CDOi participating in this training as the aggregation node.

Step 3: Key generation. In the global epoch *t*, TA honestly executes the threshold homomorphic encryption [6] algorithm to generate the homomorphic encryption key and public key.

TA randomly selects two strong prime numbers, p and q, and computes $n = pq$. where p and q satisfy $p = 2p' + 1$ and $q = 2q' + 1$, $\gcd(n, \varphi(n)) = 1$. Let $m = p'q'$ and β be a randomly chosen element in $\mathbb{Z}_N^*$ Then TA randomly selects $(a, b) \in \mathbb{Z}_n^* \times \mathbb{Z}_n^*$ and let $g = (1+n)^a \times b^n \mathrm{mod} n^2$, $g \in \mathbb{Z}_{n^2}^*$. Let $\mathrm{SK} = \beta m$, set $\theta = L(g^{\beta m}) = a\beta m$ and PK consists of g, n, and θ. PK is sent to the blockchain via smart contracts.

Step 4: Key sharing. TA uses Shamir scheme [36] for sharing the key SK, let $a_0 = \beta m$, TA randomly selects T values $\{0, 1, 2, , \ldots\ldots, n \times m - 1\}$ from $\{a_1, a_2, \ldots\ldots, a_T\}$ and constructs a polynomial $f(x) = a_T x^T + \ldots + a_2 x^2 + a_1 x^1 + a_0$. Then, TA calculates $SK_{CDOi} = f(i) mod nm$, for the ith CDO and sends $\mathrm{SK}_{\mathrm{CDO}i}$ to the CDOs involved in training, respectively.

Step 5: Local training. In the global epoch t, each healthcare organization CDOi trains the local model $\mathrm{LM}^{(i)}$ using the local dataset *Di*. If it is the first epoch, the initial global model parameter IGM is used for training; otherwise, the local model is trained using the DGM from the previous round, where n_i is the total number of samples in dataset *Di* and $n = \sum_{i=1}^{M} n_i$ denotes the total number of all samples of participants. The loss function for each participant is $F_i(\omega) = \frac{1}{n_i}\sum_{r \in D_i} f_r(\omega)$, $f_r(\omega)$ is the loss of a single data point, and the model parameters are updated as follows:

$$\omega_t^i = \omega_{t-1}^i - \eta_i \nabla F_i(\omega_{t-1}^i), \tag{1}$$

$$\mathrm{LM}^{(i)} = \frac{n_i}{n} \omega_t^i \ , \tag{2}$$

where ω_{t-1}^i denotes the initial model parameter IGM or the global model parameter DGM decoded in the previous round, η_i denotes the learning rate of the *i*th participant, and $\nabla F_i(\omega_{t-1}^i)$ denotes the gradient of the *i*th participant.

After the training is completed, CDOi encrypts the trained $\mathrm{LM}^{(i)}$ using PK, randomly selecting $x_i \in \mathbb{Z}_N^*$ and computing the ciphertext.

$$\mathrm{ELM}^{(i)} = E\left(\mathrm{LM}^{(i)}\right) = g^{\mathrm{LM}^{(i)}} {x_i}^n \mathrm{mod} n^2. \tag{3}$$

After that, $\mathrm{ELM}^{(i)}$ is uploaded to the IPFS file system, and then the hash returned by IPFS and other related data are added as transactions and uploaded to the blockchain.

Step 6: Model parameter aggregation. After waiting for a preset time t_0, the aggregation node aggregates all encrypted local model parameters $\mathrm{ELM}^{(i)}$, obtains the EGM and uploads it to the IPFS file system, and then adds the hash returned by IPFS and other related data as transactions and uploads them to the blockchain. The aggregation process is as follows:

$$\mathrm{EGM} = \prod_{i=1}^{M} \mathrm{ELM}^{(i)} = g^{\Sigma \mathrm{LM}^{(i)}} \left(\prod x_i\right)^n \mathrm{mod} n^2 (1 \le i \le M) \ . \tag{4}$$

Step 7: Partial decryption of model parameters. each CDOi gets EGM from the blockchain and partially decrypts it using $\mathrm{SK}_{\mathrm{CDO}i}$:

$$\mathrm{PGM}^{(i)} = \mathrm{EGM}^{2M! SK_{\mathrm{CDO}i}} \mathrm{mod} n^2. \tag{5}$$

After that, the decrypted $\mathrm{PGM}^{(i)}$ is uploaded to IPFS, and then the hash value returned by IPFS is uploaded to the blockchain through a smart contract.

Step 8: Final decryption: the aggregation node collects the $\mathrm{PGM}^{(i)}$ uploaded by CDO*i*, if the collected $\mathrm{PGM}^{(i)}$ is less than the threshold T then the final decryption result cannot be obtained, i.e., the aggregation update of this round, otherwise let S be the set with at least T $\mathrm{PGM}^{(i)}$, we call the partial decryption in S as $\mathrm{PGM}^{(j)}$ and obtain the final decryption result DGM in the following way:

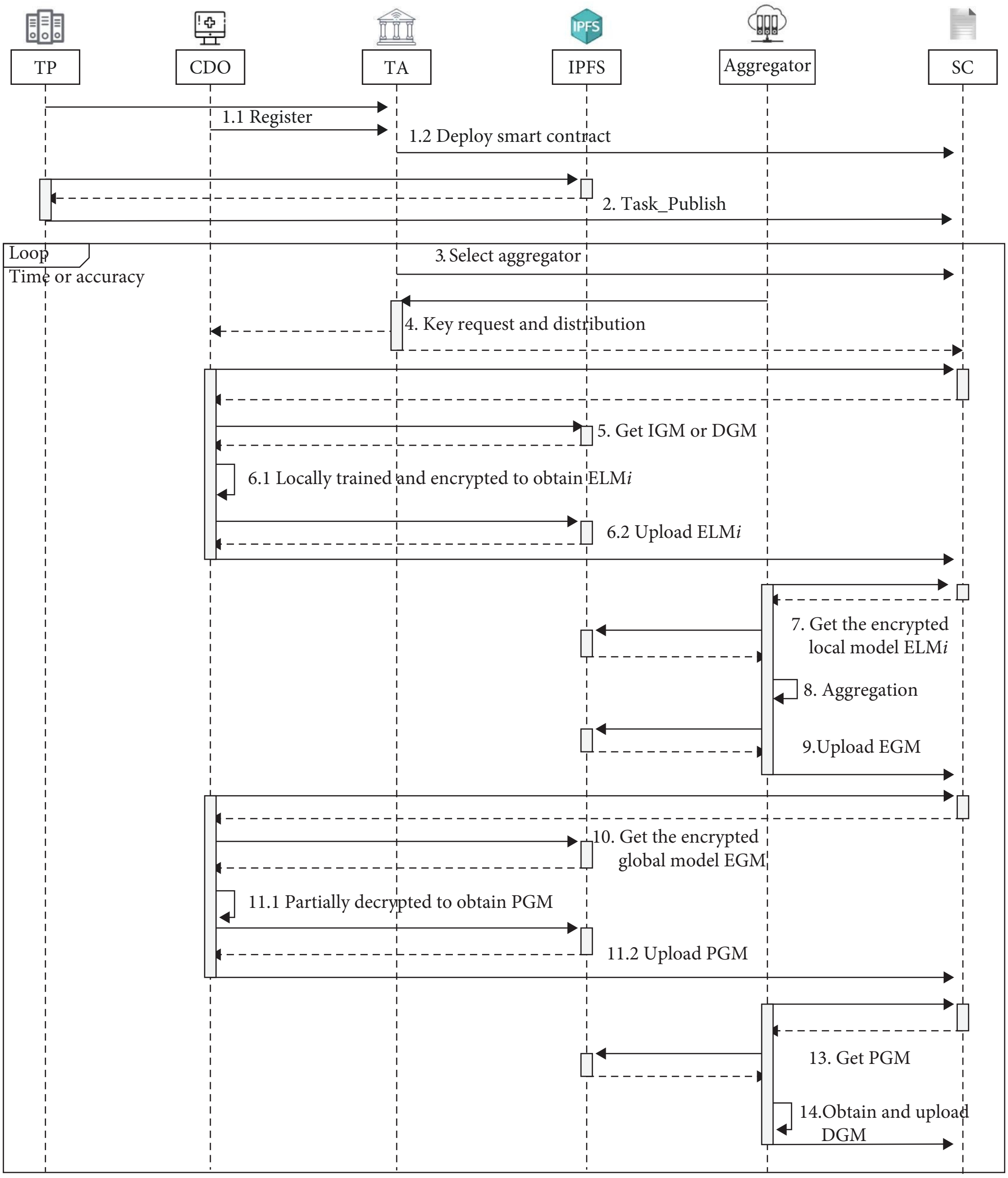

TP
CDO
TA
IPFS
IPFS
Aggregator
SC
1.1 Register
1.2 Deploy smart contract
2. Task_Publish
Loop
Time or accuracy
3. Select aggregator
4. Key request and distribution
5. Get IGM or DGM
6.1 Locally trained and encrypted to obtain ELMi
6.2 Upload ELMi
7. Get the encrypted local model ELMi
8. Aggregation
9.Upload EGM
10. Get the encrypted global model EGM
11.1 Partially decrypted to obtain PGM
11.2 Upload PGM
13. Get PGM
14.Obtain and upload DGM

Figure 2: System timing diagram.

$$\mathrm{DGM}=\sum_{i=1}^{M}\mathrm{LM}^{(i)}=L\left(\prod_{j\in S}\left(\mathrm{PGM}^{(j)}\right)^{2\mu_j^S}\mathrm{mod}\,\mathrm{n}^2\times\right)\left(4(M!)^2\theta\right)^{-1}\mathrm{mod}\,\mathrm{n}, \tag{6}$$

where $L(x)=\frac{x-1}{n}$, $\mu_j^S=M!\times\prod_{j'\in S\{j\}}\frac{j'}{j'-j}\in\mathbb{Z}$.

Step 9: Model update. CDOi fetches the latest global model parameters from the blockchain and updates the local model parameters for training.

The above process is repeated until the model converges, or reaches the required accuracy, or reaches the set number of training rounds.

4. Evaluations and Discussion

To evaluate the effectiveness of the solution, we conducted local simulations. The simulation experiments were conducted under Windows 11 system environment with the hardware configuration of AMD R7-5800H CPU, RTX3060 GPU, and 16 GB RAM. Federated learning collaborative training is implemented using the PyTorch machine learning framework, and the training model is a logistic regression model that is widely used for disease prediction. For tabular data, the diabetes dataset [37] and the breast cancer dataset [38] are chosen for the experimental test dataset, and for image data, the MNIST dataset [39] is used for the test dataset,

```
Input: $\mathrm{Add}_{\mathrm{CDO}}$
Output: $\mathrm{Agg}_{\mathrm{CDO}}$
procedure confirm()
if confirm trans.sender is not CDO then
   throw
else
   Save $\mathrm{Add}_{\mathrm{CDO}}$ to $\mathrm{Add}_{\mathrm{CDO}}[]$
end if
end procedure
procedure select()
if select trans.sender is not TA then
   throw;
else
   $r =$ random (block.timestap, block.diffculty, CDO.length)
   select. $\mathrm{Agg}_{\mathrm{CDO}} = \mathrm{Add}_{\mathrm{CDO}}[r]$
end if
end procedure
```

Algorithm 1: SC.

which is widely used in machine learning, and the above 80% of the data in the dataset is used for training, and 20% of the data are used for testing. In addition, the part involving smart contracts is processed using Solidity language. In order to analyze the practicality of the solution in this paper, the smart contracts designed in this paper are written and debugged on REMIX IDE, and the Ethernet blockchain is built locally by Truffle and Ganache.

In the experimental section, the scheme of this paper is compared with those of literatures [24, 28], which are both blockchain and federated learning-based medical data sharing schemes, where literature [24] uses smart contract technology and literature [28] uses differential privacy technology, respectively.

4.1. Accuracy. The dataset used in this article will be distributed across three different CDOs, and their respective medical data will be shared through federated learning. As shown in Figures 3 and 4, the accuracy and corresponding loss curves of the diabetes dataset, breast cancer dataset and MNIST dataset after three different CDO trainings are respectively shown. From Figure 3, it can be seen that the accuracy of the global model trained by the federation gradually exceeds that of a single CDO as the number of iterations increases. This is because as the number of CDOs increases, the total data volume also increases, resulting in an increase in the final accuracy of the global model. Moreover, compared to machine learning, federated learning has better privacy. As shown in Figure 4, the overall training process gradually stabilizes after 50 epochs. Generally speaking, as the loss value of each CDO training decreases, its corresponding accuracy will continue to increase. However, after a certain epoch, the accuracy tends to a constant value, but the loss will continue to decrease until it also tends to a constant value, and the training process tends to stabilize.

In addition, as shown in Figure 5, the scheme in this paper chooses to use homomorphic encryption to continue to enhance the privacy of federated learning, and by comparison, the accuracy of the scheme in this paper using threshold homomorphic encryption is essentially the same as the scheme in the literature [24] that does not use privacy enhancement methods, and is more accurate than the scheme in the literature [28] that uses differential privacy, so the homomorphic encryption used in this paper method used in this paper does not affect the federated learning accuracy and will further enhance the privacy of federated learning.

4.2. Time Performance. In the scheme of this paper, the use of threshold homomorphic cryptography, as well as blockchain technology, adds additional time consumption compared to traditional federated learning. For each of these two techniques, this paper presents an analysis.

4.2.1. Homomorphic Encryption. Compared with the schemes in the literatures [24, 28], the main time consumption of the scheme in this paper partly comes from the use of threshold homomorphic encryption technique, which will increase the time consumption to a certain extent, but the security will be improved to a greater extent, please see Section 4.5.2 for detailed security proof.

As shown in Table 1, the encryption and decryption time of the used threshold Paillier encryption system is measured in this paper. From Table 1, it is easy to see that the time consumption of the homomorphic encryption technique is related to the size of the data, and the larger the data volume of a single piece of data, the more time resources it consumes. For example, the MNIST dataset, which can be represented by a 2D array of 28×28 with a total of 784 feature values for one of the images, has a large data volume, so the encryption and decryption process takes longer time, while the Diabetes dataset has less number of feature values, so the encryption and decryption time consumption is less.

The encryption process of the homomorphic encryption scheme used in this paper takes less time, while in the decryption process, the threshold Paillier encryption system requires a higher number of CDOs than the threshold for partial decryption, and then the aggregator for final decryption, so it takes a longer time. Although the decryption process of the threshold Paillier system takes longer, it solves the problem of insufficient privacy caused by a single pair of keys in the application of the original Paillier system to federated learning and improves the overall privacy and robustness of the federated learning system. In addition, the key length of the scheme used in this paper is 1,024 when encrypting the model parameters.

4.2.2. Throughput Rate. Another part of the time consumption comes mainly from the use of blockchain technology. This paper uses the ethernet platform to build smart contracts, but the TPS (transaction per second) of the ethernet platform is low, only about 14 TPS, which greatly limits the efficiency of this paper. In a practical deployment, CDOs can jointly maintain a federated chain, which will use a consensus mechanism that is more efficient than the Ethernet platform

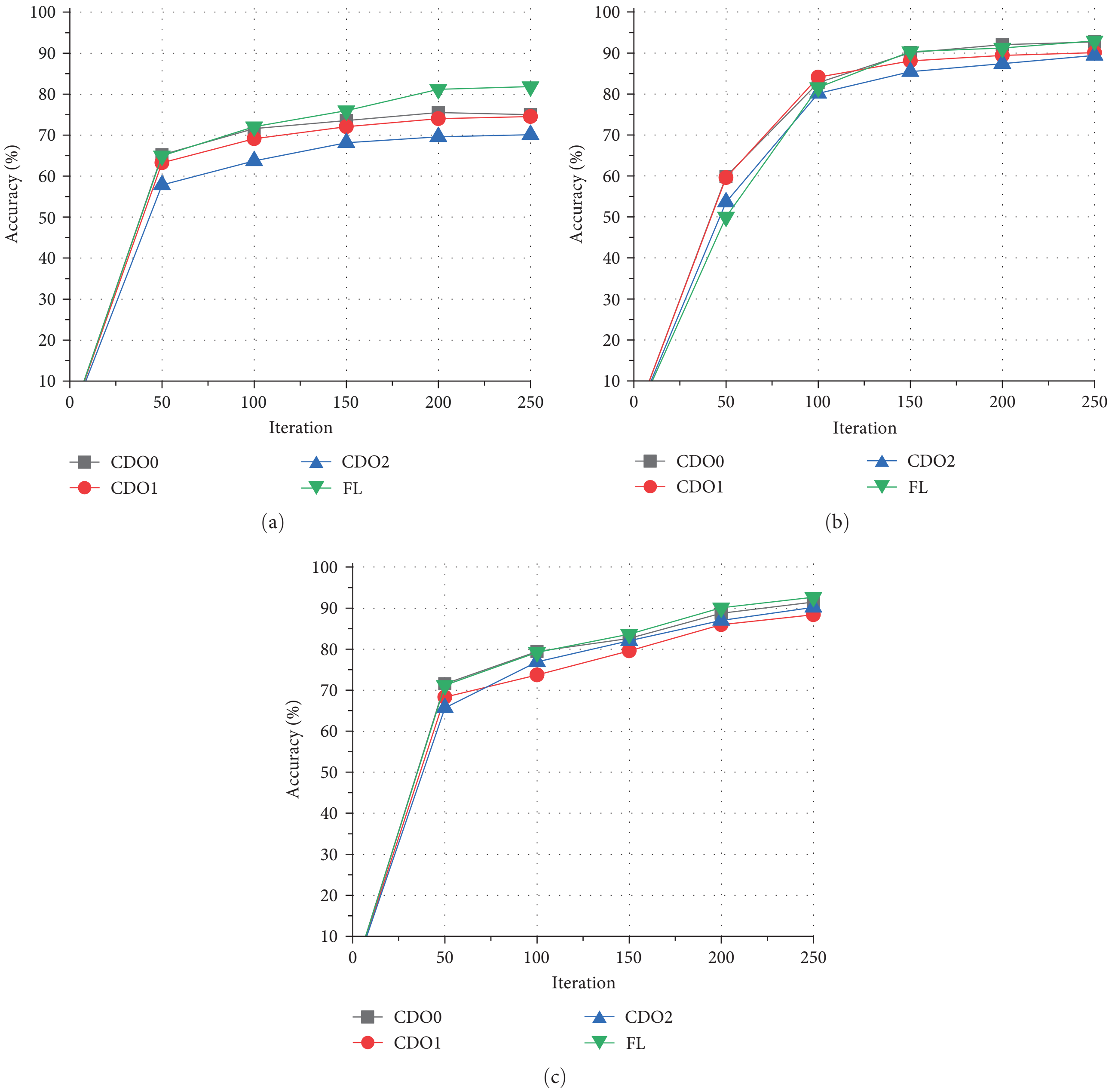


Figure 3: CDO*i* accuracy in diabetes dataset, breast_cancer dataset, and MNIST dataset: (a) accuracy of diabetes dataset; (b) accuracy of breast_cancer dataset; (c) accuracy of MNIST.

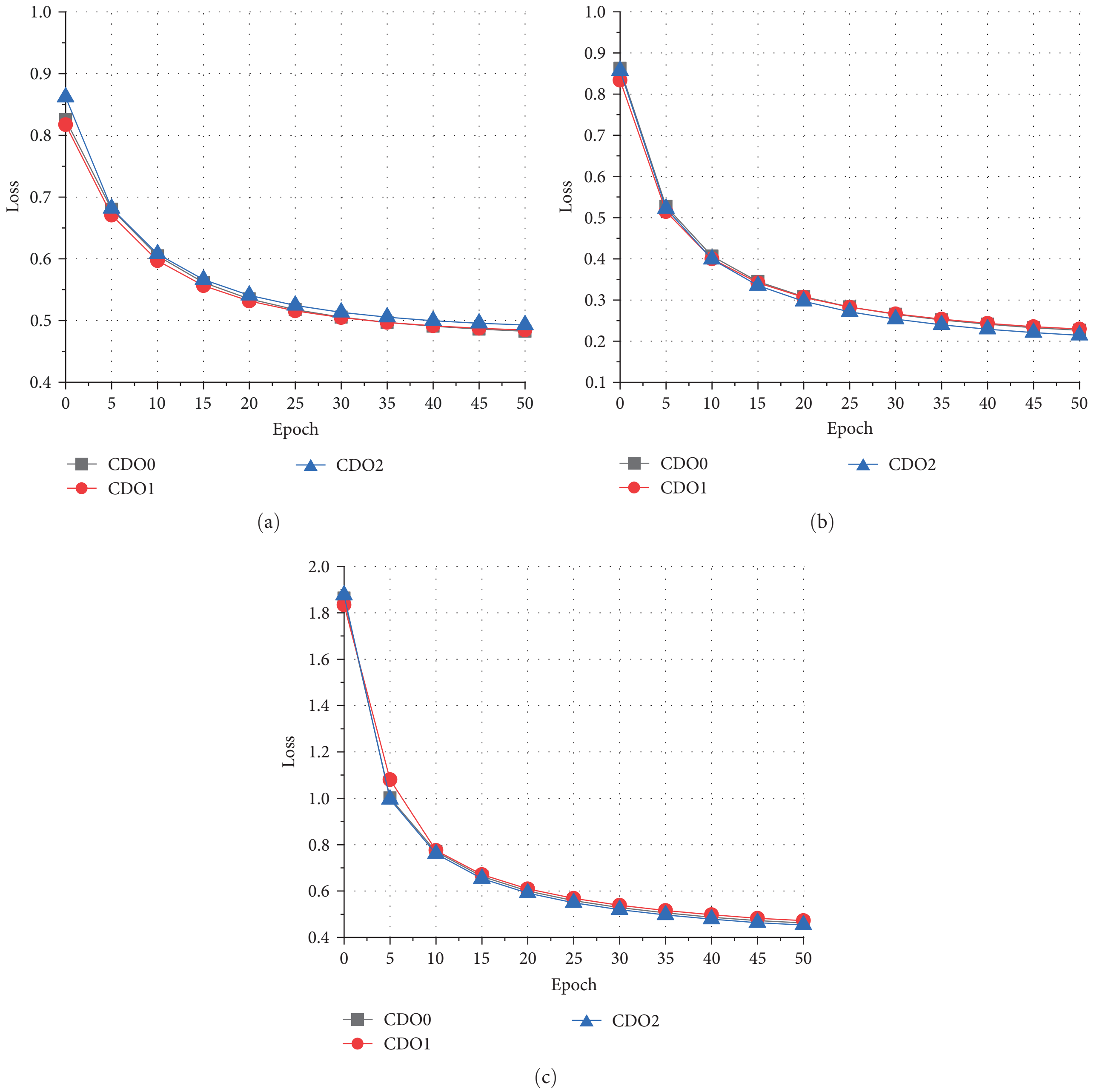
and can achieve higher throughput rates. In a blockchain that can use smart contracts, TPS is mainly related to the $\text{Gas}_{\text{limit}}$ of a single block, the block time interval $\text{Block}_{\text{time}}$ and the gas Tx_{Gas} required to calculate the execution of the transaction, and we can calculate TPS by the following formula:

$$\text{TPS} = \frac{\text{Gas}_{\text{limit}}}{Tx_{\text{Gas}} \times \text{Block}_{\text{time}}}. \quad (7)$$

In order not to affect the synchronization speed of the block, the generation interval of the block is set to 5 s and $\text{Gas}_{\text{limit}}$ is 0x8ffffff. We take UploadModel() and SelectAgg() as functions as examples; the TPS of both can reach 270TPS and 226TPS, respectively, which can meet the application requirements of the medical federated learning framework in this paper.

4.3. Blockchain Consumption. At present, there are two main forms of blockchain-based federated learning frameworks: one is based on existing blockchain platforms, such as Ethereum, and the other is to design the corresponding blockchain by itself according to the application scenario and both approaches have their own advantages and disadvantages.

In order to more intuitively reflect the resource consumption of using blockchain, the Ether platform is chosen as the experimental platform to evaluate the actual resource



(a)

(b)

(c)

Figure 4: CDO*i* Loss in diabetes dataset, breast_cancer dataset, and MNIST dataset: (a) loss of diabetes dataset; (b) loss of breast_cancer dataset; (c) loss of MNIST.

consumption through the consumption of gas. The deployment and invocation of smart contracts need to consume gas, and the consumption of gas is closely related to the amount of data processing and the complexity of algorithms, so the consumption of gas can reflect the resource consumption of using smart contracts. In addition, there is a bottleneck in the storage of blockchain. Currently, the block size of Bitcoin is generally fixed at 1 MB, and the block size of Ether is generally no more than 2 kB, so the storage of the blockchain is an issue that needs to be considered.

Table 2 shows the comparison between the scheme of this paper and the scheme of the literatures [24, 28]. The literature [24] chooses to use smart contracts to implement the local model aggregation function, which consumes 1,521,269 gas, while the literature [28] does not use smart contracts, while the gas consumption of the scheme in this paper is 979,804, which is lower than the former. In addition, in terms of time complexity, literature [24] implements complex aggregation function through smart contracts, which requires multiple rounds of cyclic operations, so the time complexity is higher; literature [28] combines the aggregation process with the blockchain consensus process, and the selection of aggregators is performed through the committee in the consensus process, so the time complexity is lower,

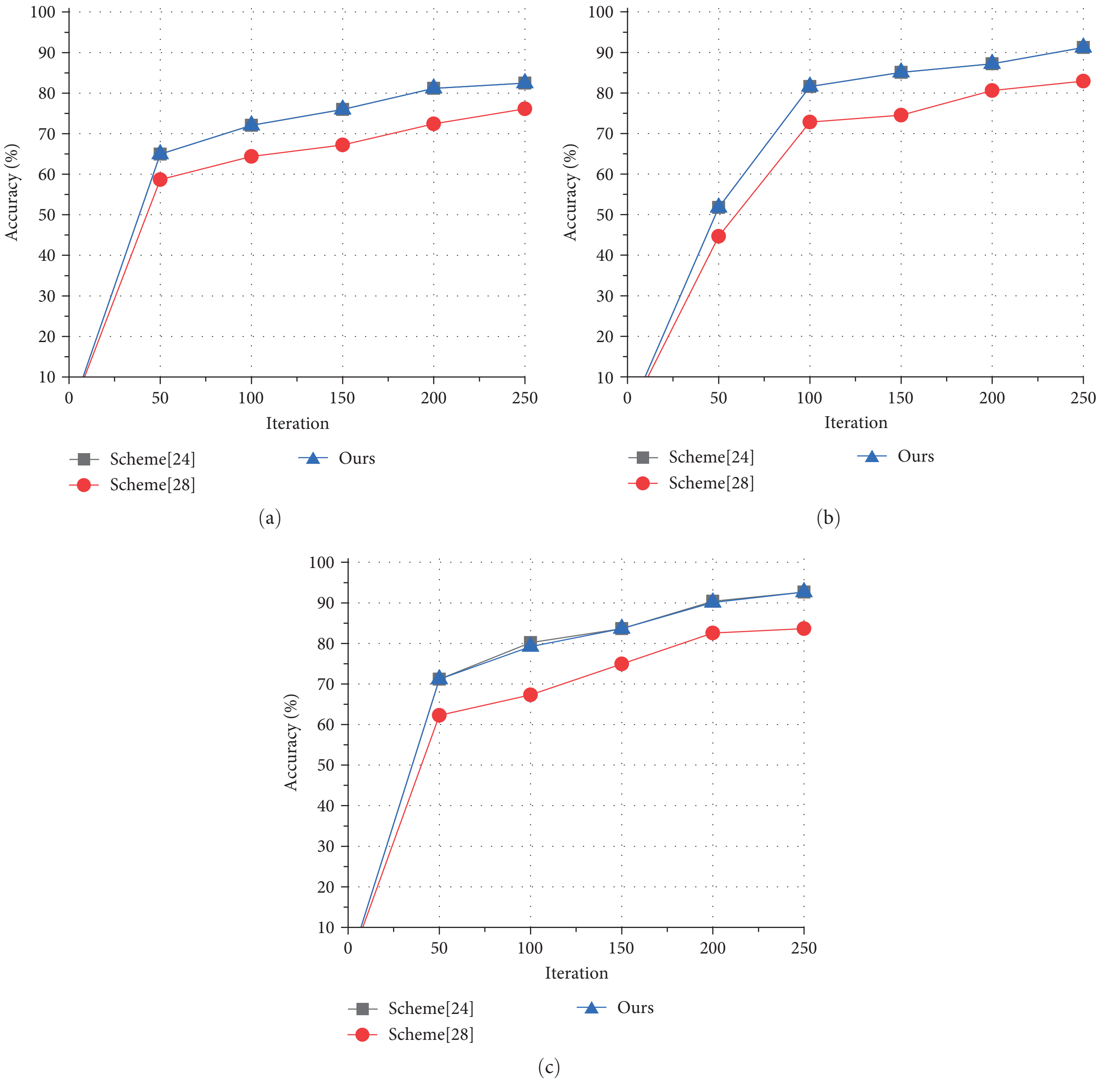


Figure 5: Comparison of accuracy rates of different solutions: (a) accuracy of diabetes dataset; (b) accuracy of breast_cancer dataset; (c) accuracy of MNIST.

Table 1: Encryption/decryption time consumption.

Dataset	Encryption time (ms)			Decryption time (ms)
	CDO0	CDO1	CDO2	Threshold paillier
Diabetes	2.257	1.981	2.091	66.835
Breast_cancer	1.219	1.284	1.378	44.876
MNIST	1.861×10^2	1.766×10^2	1.971×10^2	6.181×10^3

while the scheme in this paper only requires simple selection operations through smart contracts that are more in line with the current specification of smart contract usage. After using IPFS for storage, the data to be stored for a single transaction submitted by a smart contract is no longer the local model parameters but the hash corresponding to the local model parameters, with a fixed size of 32 B, which will consume less than the previous two schemes. Moreover, if the CNN model is used for federated learning training, hundreds of thousands or even millions of model parameters will be

Table 2: Blockchain performance comparison.

Paper	Smart contract	Time complexity	Individual transaction data
[24]	Aggregation	$O\ (n^m)$	Size (modal)
[28]	—	$O\ (n)$	Size (modal)
Our scheme	Aggregate node selection	$O\ (n)$	Size (hash)

Table 3: Comparison of healthcare data-sharing solutions.

Paper	Blockchain	Smart contract	Methods
[8]	—	—	—
[24]	Y	Y	—
[28]	Y	—	Differential privacy
Our scheme	Y	Y	Homomorphic encryption

generated, at which time the blockchain will not be able to meet the storage requirements of the scheme in literature [24] while the scheme in this paper does not need to consider the storage requirements of the blockchain too much, so the scheme in this paper has better pervasiveness.

4.4. Program Comparison. As shown in Table 3, existing techniques have used federated learning to address healthcare data privacy, but little work has been done to simultaneously use blockchain and smart contracts for federated learning to improve security. Furthermore, the privacy of federated learning can continue to be enhanced by using additional privacy-preserving methods, but the noise added by differential privacy can affect the usability of the final model when dealing with medical data, while homomorphic encryption does not. Overall, the scheme in this paper makes medical data more secure and private during sharing by using smart contracts as well as homomorphic encryption.

4.5. Program Evaluation

4.5.1. Proof of Correctness. In this paper, we protect the privacy of the parameter passing process by using a threshold Paillier cryptosystem for key generation, where TA constructs a polynomial to share the key among CDOs using Shamir secret sharing. We consider the worst-case scenario where only T CDOs have uploaded the correct part of decryption in the blockchain. Let S be the set of T correct partial decryptions and S be a subset $j \in S$ of M, such that $s_j = \mathrm{SK}_{\mathrm{CDO}i}$ and the partial decryption is $\mathrm{PGM}^{(j)} = \mathrm{EGM}^{2M!s_j}\mathrm{modn}^2$.

By using the Lagrangian interpolation formula, we can obtain the following:

$$M!f(0) = M!m\beta = \sum_{j\epsilon S} \mu_j^S f(j)\mathrm{modnm}. \tag{8}$$

Thus, we can obtain the following:

$$\mathrm{EGM}^{4(M!)^2 m\beta} = \prod_{j\epsilon S} \mathrm{EGM}^{4M!s_j\mu_j^S} = \prod_{j\in S} \left(\mathrm{PGM}^{(j)}\right)^{2\mu_j^S}\mathrm{modn}^2. \tag{9}$$

From the homomorphism of the Paillier cryptosystem, it follows that

$$E\left(\mathrm{LM}^{(1)} + \mathrm{LM}^{(2)} + \ldots + \mathrm{LM}^{(M)}\right) = E\left(\mathrm{LM}^{(1)}\right) \times E\left(\mathrm{LM}^{(2)}\right) \times \ldots \times E\left(\mathrm{LM}^{(M)}\right). \tag{10}$$

Next, calculate the following:

$$\mathrm{EGM} = \prod_{i=1}^{M} \mathrm{ELM}^{(i)} = g^{\sum \mathrm{LM}^{(i)}} \left(\prod x_i\right)^n \mathrm{modn}^2. \tag{11}$$

Thus

$$\mathrm{EGM}^{4(M!)^2 m\beta} = g^{\sum_{i=1}^{M} \mathrm{LM}^{(i)} 4(M!)^2 m\beta} \times \left(\prod_{i=1}^{M} x_i\right)^{n4(M!)^2 m\beta} \mathrm{modn}^2. \tag{12}$$

Ultimately, we can get the following:

$$\mathrm{DGM} = \sum_{i=1}^{M} \mathrm{LM}^{(i)} = L\left(\prod_{j\in S} \left(\mathrm{PGM}^{(j)}\right)^{2\mu_j^S}\mathrm{modn}^2 \times\right)\left(4(M!)^2\theta\right)^{-1}\mathrm{modn}. \tag{13}$$

The aggregator can use no less than T correct partial decryptions to recover the local model parameters of the aggregation at the time of final decryption.

4.5.2. Proof of Security. In the federated learning framework proposed in this paper, each medical institution CDO encrypts the trained local model parameters, uploads them to IPFS, and later uploads the hash value returned from IPFS to the blockchain. In this process, all users on the blockchain can obtain the hash value from the blockchain and obtain the encrypted local parameter model $\mathrm{ELM}^{(i)}$ from IPFS based on the hash value, and since the threshold, Paillier homomorphic cryptosystem is semantically secure, and each CDO does not have a complete key; therefore, CDOs cannot obtain private data based on $\mathrm{ELM}^{(i)}$, so the medical federated learning framework proposed in this paper can protect the privacy of local model parameters during the federated learning training process.

Theorem 1. *Under the DCRA (decisional composite residuosity assumption), the medical federated learning framework proposed in this paper can protect the privacy of local model parameters.*

Proof. The security of the medical federated learning framework proposed in this paper mainly depends on the threshold Paillier homomorphic cryptosystem used, and if the system is semantically secure, the medical federated learning framework proposed in this paper is semantically secure. □

Suppose there exists an adversary $\mathscr{A}$ that can break the semantic security of the threshold Paillier cryptosystem and thus obtain private data. Given a challenger, it can use $\mathscr{A}$ to break the semantic security of the original Paillier.

Step 1: $\mathscr{A}$ Select the server that has compromised T CDOs and has access to the private data of the compromised server.

Step 2: The challenger first obtains the public key PK of the Paillier cryptosystem and randomly selects $(a_1, b_1, \theta) \in \mathbb{Z}_n^* \times \mathbb{Z}_n^* \times \mathbb{Z}_n^*$. Then set $g_1 = g^{a_1} \times {b_1}^n \text{modn}^2$. Meanwhile, the attacker randomly selects $s_1, s_2, \ldots s_T$ from $\{0, \ldots, \lfloor n^2/4 \rfloor\}$ and sends $\{n, g_1, \theta, s_1, s_2, \ldots s_T\}$ to the adversary $\mathscr{A}$.

Step 3: $\mathscr{A}$ selects the message M and sends it to the challenger, who computes the valid encryption $c = {g_1}^M x^n \text{modn}^2$ for M. Partial decryption of the corrupted CDO server can be computed as follows:

$$c_i = c^{2M!SK_{\text{CDO}i}} \text{modn}^2 (1 \leq i \leq T). \tag{14}$$

Partial decryption of other CDOs can be obtained by interpolation.

$$c_i = (1 + 2M\theta n)^{\mu^S_{(i,0)}} \times \prod_{j \in S \backslash \{0\}} c^{2\text{SK}_{\text{CDO}i}\mu^S_{(i,j)}} \text{modn}^2. \tag{15}$$

Step 4: Adversary $\mathscr{A}$ selects and sends two messages M_0 and M_1 to the challenger. Subsequently, the cryptographic prediction machine selects a random message b and sends a ciphertext c of Mb to the challenger. the challenger computes $c' = c^{a_1} \text{modn}^2$ and sends it to adversary $\mathscr{A}$.

Step 5: The adversary $\mathscr{A}$ continues to follow the stage 1 adaptive query after receiving the challenge ciphertext c'.

Step 6: The adversary $\mathscr{A}$ outputs b', where b' is a guess of the challenger. In summary, the advantage that the challenger can successfully break the semantic security of the Paillier cryptosystem is the same as the advantage that the adversary $\mathscr{A}$ has in winning the security contest.

From the literature [40], it is clear that the original Paillier cryptosystem is semantically secure under the assumption of DCRA, so no adversary $\mathscr{A}$ satisfies the condition that the threshold Paillier cryptosystem is also semantically secure.

Theorem 2. *Assuming that the CDOs involved in the federated learning task are honest but curious and that at most CDOs up to a threshold T are allowed to conspire, then the scheme in this paper will guarantee the privacy of the local model parameters of each CDO.*

Proof. Suppose there exists an attack algorithm in which an aggregator selected by a smart contract can only collude with at most $T-2$ CDOs involved in this training round. (PK, T, M) as the input to the algorithm can be obtained from the blockchain by all the CDOs participating in this training task, and therefore, $\text{LM}^{(i)}$ can be derived by the aggregator from the above input parameters and make them public. However, in the scheme of this paper, the $\text{LM}^{(i)}$ of each CDO is encrypted into ciphertext, and the original key can be recovered and decrypted only after obtaining more than a threshold T CDOs, which contradicts the assumption. Therefore, the hypothetical attack algorithm does not exist. The following example demonstrates the correctness of the theorem. □

Given two honest medical institutions CDO1 and CDO2, and a curious medical institution CDO3. CDO1 and CDO2, which have their local model parameters, $\text{ELM}^{(1)}$ and, $\text{ELM}^{(2)}$, respectively, assume the existence of an attack algorithm such that CDO3 can computationally derive the local model parameters of CDO1 and CDO2 by (PK, T, M). To verify whether CDO3 can derive the uploaded local model parameters, this scheme first uploads $\text{ELM}^{(1)}$ and $\text{ELM}^{(2)}$ to the blockchain and obtains them by CDO3, after obtaining $\text{ELM}^{(1)}$ and $\text{ELM}^{(2)}$ CDO3 derives them, after that CDO1 and CDO2 exchange the $\text{ELM}^{(1)}$ and $\text{ELM}^{(2)}$ they own, i.e., CDO1 holds $\text{ELM}^{(2)}$ and CDO2 holds $\text{ELM}^{(1)}$, and finally execute the secure training scheme in this paper. Based on the security and privacy of the scheme in this paper, each CDO only owns part of the decryption key $\text{SK}_{\text{CDO}i}$, and there is no CDO with more than a threshold T for collusion, so CDO3 cannot judge whether the input values of CDO1 and CDO2 change during the execution of the scheme in this paper and still considers that CDO1 holds $\text{ELM}^{(1)}$ and CDO2 holds $\text{ELM}^{(2)}$, but at this time the input values have changed, which contradicts the assumption, so there is no such attack algorithm. From this, it can be concluded that the scheme in this paper can guarantee the privacy of the local model parameters of each CDO when the CDO collusion does not exceed the threshold T.

In summary, first, this paper improves the privacy of healthcare data sharing among healthcare organizations by using cross-silo federated learning, where each CDO interacts with the other with model parameters rather than raw data. Second, in the attack model of malicious CDOs, malicious CDOs can obtain ciphertexts as well as partially decrypted ciphertexts directly from the blockchain because the threshold Paillier cryptosystem used in this paper is semantically secure, so the attacker must break the cryptosystem to obtain private data. In addition, the original key can be recovered only when CDOs exceed the threshold T collude, which cannot happen in reality because each CDO involved in the training guarantees the security of its data.

4.5.3. *Antiattack Type.*

(1) Resistance to inference attacks. In the blockchain-based federated learning framework, the local model parameters are uploaded to the blockchain network, where all nodes in the network can access the model parameters and reason backward to the original data based on the model. For such attacks, the scheme in this paper uses the threshold Paillier cryptosystem to encrypt the local model parameters and broadcast them to the blockchain network in the ciphertext and be recorded in the distributed ledger.

(2) Anti-single-point-of-failure. By storing the local model parameters through the blockchain, we prevent data loss during the training process and use a smart contract to select the CDO participating in the training as the aggregator instead of a fixed central server or task publisher as the aggregator, thus solving the possible single point of failure in the model aggregation process. If an attacker wants to disrupt the decryption process, he needs to attack at least $M - T - 2$ nodes simultaneously to interrupt the decryption process.

(3) Anticollusion attack. The scheme in this paper adopts a threshold homomorphic encryption system. When the number of nodes performing collision is less than the threshold T, the private keys of the colluding nodes cannot be recovered by Lagrangian interpolation, and the colluding nodes cannot obtain the relevant information of the local model.

4.5.4. Auditability. This paper provides a complete record of the flow and use of data in the federated learning training process through blockchain and IPFS. The open, transparent, and tamper-evident nature of blockchain is used to ensure the integrity of data use records, improve the reliability of federated learning, and can provide data support for the design of incentives.

4.5.5. Fairness. In the scheme of this paper, since there is no incentive mechanism involved and no special reward for completing the work, to make each CDO contribute their arithmetic power fairly, this paper designs a smart contract to randomly select aggregation nodes for local model parameter aggregation from the CDOs participating in this training, instead of using a fixed aggregator for aggregation, to achieve higher fairness and stronger security.

5. Conclusions

To address the problem of secure sharing of medical privacy data among medical institutions, this paper proposes a medical data privacy protection framework by combining blockchain and cross-silo federated learning. The framework in this paper uses cross-silo federated learning to build a collaborative training platform for multiple medical institutions to facilitate the flow of medical data and store intermediate parameters through blockchain to prevent the loss of intermediate parameters and enhance the trust among medical institutions. Using smart contracts to select aggregation nodes instead of fixed servers for aggregation prevents the single point of failure problem in the federated learning process. In addition, the use of threshold homomorphic encryption solves the inference attacks that may occur during the training process and addresses the limitation of using single-key homomorphic encryption in federated learning. The privacy protection framework of medical data proposed in this paper breaks the “data silo” between different medical institutions, solves the privacy and security problems of traditional federated learning, and realizes the secure sharing of medical data while protecting user privacy.

Symbols

M:	Number of CDOs participating in training
Di:	Dataset of the ith CDO
T:	The threshold value in the threshold homomorphic encryption scheme
SK:	The private key in the original Paillier homomorphic encryption scheme
SK_{CDOi}:	Threshold homomorphic encryption private key for the ith CDO
Add_{CDO}:	The address of the CDO
Agg_{CDO}:	The address of the aggregator
PK:	Threshold homomorphic encryption public key
IGM:	Initial global model parameters
$LM^{(i)}$:	Local model parameters of the ith CDO
$ELM^{(i)}$:	Cryptographic local model parameters for the ith CDO
EGM:	Cryptographic global model parameters after aggregation
$PGM^{(i)}$:	Global model parameters for the ith CDO partial decryption
DGM:	Global model parameters after decryption.

Data Availability

The medical data used to support the results of this study can be found at https://datahub.io/machine-learning/diabetes and https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(original).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is funded by The National Key Research and Development Program of China (2017YFB0802300) and the Natural Science Foundation of Hebei Province of China (F2021201052).

References

[1] J. Konečný, H. B. McMahan, D. Ramage, and P. Richtárik, “Federated optimization: distributed machine learning for on-device intelligence,” arXiv preprint arXiv: 1610.02527, 2016.

[2] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: strategies for improving communication efficiency," arXiv preprint arXiv: 1610.05492, 2016.

[3] P. Kairouz, H. B. McMahan, B. Avent et al., "Advances and open problems in federated learning," *Foundations and Trends® in Machine Learning*, vol. 14, no. 1-2, pp. 1–210, 2021.

[4] D. Yaga, P. Mell, N. Roby, and K. Scarfone, "Blockchain technology overview," arXiv preprint arXiv:1906.11078, 2019.

[5] Z. Zheng, S. Xie, H.-N. Dai et al., "An overview on smart contracts: challenges, advances and platforms," *Future Generation Computer Systems*, vol. 105, pp. 475–491, 2020.

[6] P. A. Fouque, G. Poupard, and J. Stern, "Sharing decryption in the context of voting or lotteries," in *Financial Cryptography*, vol. 1962 of *Lecture Notes in Computer Science*, pp. 90–104, Springer, Berlin, Heidelberg, 2000.

[7] J. Benet, "Ipfs-content addressed, versioned, p2p file system," arXiv preprint arXiv: 1407.3561, 2014.

[8] T. S. Brisimi, R. Chen, T. Mela, A. Olshevsky, I. C. Paschalidis, and W. Shi, "Federated learning of predictive models from federated electronic health records," *International Journal of Medical Informatics*, vol. 112, pp. 59–67, 2018.

[9] S. Silva, B. Gutman, E. Romero, P. M. Thompson, A. Altmann, and M. Lorenzi, "Federated learning in distributed medical databases: meta-analysis of large-scale subcortical brain data," in *2019 IEEE 16th international symposium on biomedical imaging (ISBI 2019)*, pp. 270–274, IEEE, 2019.

[10] M. J. Sheller, B. Edwards, G. A. Reina et al., "Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data," *Scientific Reports*, vol. 10, Article ID 12598, 2020.

[11] W. Zhang, T. Zhou, Q. Lu et al., "Dynamic-fusion-based federated learning for covid-19 detection," *IEEE Internet of Things Journal*, vol. 8, no. 21, pp. 15884–15891, 2021.

[12] N. Rieke, J. Hancox, W. Li et al., "The future of digital health with federated learning," *NPJ Digital Medicine*, vol. 3, no. 1, pp. 1–7, 2020.

[13] E. Darzi, F. Dubost, N. M. Sijtsema, and P. M. A. van Ooijen, "The hidden adversarial vulnerabilities of medical federated learning," ar**v preprint ar**v: 2310.13893, 2023.

[14] L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," in *2019 IEEE symposium on security and privacy (SP)*, pp. 691–706, IEEE, 2019.

[15] B. Hitaj, G. Ateniese, and F. Perez-Cruz, "Deep models under the GAN: information leakage from collaborative deep learning," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 603–618, ACM, 2017.

[16] R. Shokri and V. Shmatikov, "Privacy-preserving deep learning," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pp. 1310–1321, IEEE, Monticello, IL, USA, 2015.

[17] X. Wu, Y. Zhang, M. Shi, P. Li, R. Li, and N. N. Xiong, "An adaptive federated learning scheme with differential privacy preserving," *Future Generation Computer Systems*, vol. 127, pp. 362–372, 2022.

[18] F. Wang, H. Zhu, R. Lu, Y. Zheng, and H. Li, "A privacy-preserving and non-interactive federated learning scheme for regression training with gradient descent," *Information Sciences*, vol. 552, pp. 183–200, 2021.

[19] J. Park and H. Lim, "Privacy-preserving federated learning using homomorphic encryption," *Applied Sciences*, vol. 12, no. 2, Article ID 734, 2022.

[20] O. T. Tawosa, *Harnessing the power of distributed computing: advancements in scientific applications, homomorphic encryption, and federated learning security*, University of Nevada, Reno, 2023.

[21] J. Ma, S.-A. Naas, S. Sigg, and X. Lyu, "Privacy-preserving federated learning based on multi-key homomorphic encryption," *International Journal of Intelligent Systems*, vol. 37, no. 9, pp. 5880–5901, 2022.

[22] A. Qammar, A. Karim, H. Ning, and J. Ding, "Securing federated learning with blockchain: a systematic literature review," *Artificial Intelligence Review*, vol. 56, no. 5, pp. 3951–3985, 2023.

[23] L. Javed, A. Anjum, B. M. Yakubu, M. Iqbal, S. A. Moqurrab, and G. Srivastava, "ShareChain: blockchain-enabled model for sharing patient data using federated learning and differential privacy," *Expert Systems*, vol. 40, no. 5, Article ID e13131, 2023.

[24] O. El Rifai, M. Biotteau, X. de Boissezon, I. Megdiche, F. Ravat, and O. Teste, "Blockchain-based federated learning in medicine," in *Artificial Intelligence in Medicine. AIME 2020*, vol. 12299 of *Lecture Notes in Computer Science*, pp. 214–224, Springer, Cham, 2020.

[25] S. Awan, F. Li, B. Luo, and M. Liu, "Poster: a reliable and accountable privacy-preserving federated learning framework using the blockchain," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pp. 2561–2563, ACM, 2019.

[26] Q. Zhang, Q. Ding, J. Zhu, and D. Li, "Blockchain empowered reliable federated learning by worker selection: a trustworthy reputation evaluation method," in *2021 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, pp. 1–6, IEEE, Nanjing, China, 2021.

[27] H. Kim, J. Park, M. Bennis, and S.-L. Kim, "Blockchained on-device federated learning," *IEEE Communications Letters*, vol. 24, no. 6, pp. 1279–1283, 2019.

[28] H. Zhang, G. Li, Y. Zhang, K. Gai, and M. Qui, "Blockchain-based privacy-preserving medical data sharing scheme using federated learning," in *Knowledge Science, Engineering and Management*, vol. 12817 of *Lecture Notes in Computer Science*, pp. 634–646, Springer, Cham, 2021.

[29] U. Majeed, L. U. Khan, A. Yousafzai, Z. Han, B. J. Park, and C. S. Hong, "ST-BFL: a structured transparency empowered cross-silo federated learning on the blockchain framework," *IEEE Access*, vol. 9, pp. 155634–155650, 2021.

[30] H. Wang, X. Zhang, Y. Xia, and X. Wu, "An intelligent blockchain-based access control framework with federated learning for genome-wide association studies," *Computer Standards & Interfaces*, vol. 84, Article ID 103694, 2023.

[31] L. Feng, Y. Zhao, S. Guo, X. Qiu, W. Li, and P. Yu, "BAFL: a blockchain-based asynchronous federated learning framework," *IEEE Transactions on Computers*, vol. 71, no. 5, pp. 1092–1103, 2022.

[32] W. Issa, N. Moustafa, B. Turnbull, N. Sohrabi, and Z. Tari, "Blockchain-based federated learning for securing internet of things: a comprehensive survey," *ACM Computing Surveys*, vol. 55, no. 9, pp. 1–43, 2023.

[33] Y. Qu, M. P. Uddin, C. Gan, Y. Xiang, L. Gao, and J. Yearwood, "Blockchain-enabled federated learning: a survey," *ACM Computing Surveys*, vol. 55, no. 4, pp. 1–35, 2022.

[34] Y. Chen, J. Li, F. Wang et al., "DS2PM: a data sharing privacy protection model based on blockchain and federated learning," *IEEE Internet of Things Journal*, vol. 10, no. 14, pp. 12112–12125, 2023.

[35] G. Wood, "Ethereum: a secure decentralised generalised transaction ledger," Ethereum project yellow paper, 151: 1-32, 2014.

[36] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612-613, 1979.

[37] V. Sigillito, "Diabetes dataset," 1990, https://datahub.io/machine-learning/diabetes.

[38] W. H. Wolfberg, "Breast cancer dataset," 1995, https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(original).

[39] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[40] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Advances in Cryptology—EUROCRYPT '99*, vol. 1592 of *Lecture Notes in Computer Science*, pp. 223–238, Springer, Berlin, Heidelberg, 1999.