Hindawi
Security and Communication Networks
Volume 2022, Article ID 7003426, 8 pages
https://doi.org/10.1155/2022/7003426

Research Article

Mutual-Supervised Federated Learning and Blockchain-Based IoT Data Sharing

Jianwei Liu,[1] Qinyang Miao,[2,3] Xinmin Fan,[4] Xiaoding Wang,[2,3] Hui Lin,[2,3] and Yikun Huang[5]

[1]*Fujian Preschool Education College, Fuzhou, Fujian 350007, China*
[2]*College of Computer and Cyber Security, Fujian Normal University, Fuzhou, Fujian 350117, China*
[3]*Engineering Research Center of Cyber Security and Education Informatization, Fujian Province University, Fuzhou, Fujian 350117, China*
[4]*Network and Data Center, Fujian Normal University, Fuzhou, Fujian 350117, China*
[5]*Concord University College of Fujian Normal University, Fuzhou, Fujian 350117, China*

Correspondence should be addressed to Xiaoding Wang; wangdin1982@fjnu.edu.cn and Hui Lin; linhui@fjnu.edu.cn

Received 12 July 2022; Revised 3 September 2022; Accepted 5 September 2022; Published 8 October 2022

Academic Editor: Chin-Ling Chen

Due to the decentralized, tamper-proof, and auditable properties of blockchain, more and more scholars and researchers are studying the application of blockchain technology in IoT data sharing. Federated learning is an effective way to enable data sharing, but can be compromised by dishonest data owners who may provide malicious models. In addition, dishonest data requesters may also infer private information from model parameters. To solve the above problems, a secure data sharing mechanism based on mutual-supervised federated learning and blockchain, BPCV-FL, is proposed. This mechanism ensures data privacy by adopting gradient descent algorithm with differential privacy protection in local model training and ensures the reliability of shared data through mutual supervision on the blockchain. Experimental results show that the proposed BPCV-FL has high accuracy and security in IoT data sharing.

1. Introduction

With the application of new-generation communication technologies and artificial intelligence technologies such as 5G and IPV6, the Internet of Things (IoT) has also developed rapidly [1]. The Internet of Things refers to the interconnection between things and things based on computer technology and forms an intelligent network technology [2], through which data transmission [3], and data sharing and storage [4] can provide reliable intelligent management [5], including online monitoring, real-time positioning, remote alarm, and other functions. The Internet of Things has created a new world that is measurable and quantifiable [6]. In this world, every terminal has the potential to become a data generator and a data consumer, and thus generate massive amounts of data. The application of these data will provide us with more extensive and high-quality value-added services, and how to ensure the authenticity, reliability, and validity of these data and protect the privacy and security of data providers has become a major challenge in technological development.

To realize the sharing of data in the IoT world, we face the following two challenges. First, it is impossible to achieve effective high-trust relationships between individuals, institutions, and other organizations in the Internet of Things, but it is necessary to share data with each other securely and reliably. How to realize reliable data sharing in an untrusted environment is a technical problem that needs to be solved urgently. Secondly, the issue of data privacy leakage is a pain point that currently plagues various data manufacturers, and it has become another important factor restricting data sharing. More and more organizations, institutions, and individuals are refusing to share data because of the potential for their data privacy to be compromised. Therefore, before we consider sharing important data, we must prepare for data protection to protect data privacy.

As a special distributed machine learning technology, federated learning [7, 8] has received extensive attention in the industry, providing technical support for secure data sharing in distributed situations. The idea of distributed training of federated learning reduces the computational burden of centralized equipment and protects the data privacy of the data owner by aggregating the model parameters trained locally by the data owner instead of the source data, thus effectively solving the problem of data collection [9]. This provides a parallel data sharing scheme for each user, organization, or institution, and the status of each participant is equal, which realizes fair cooperation and ensures that the participants can share data safely while maintaining independence. Blockchain [10] technology can effectively solve trust problems by virtue of its decentralization, nontampering, undeniable, traceability, and other characteristics, and has broad application prospects. At present, blockchain technology has become a research hotspot in data sharing.

According to above analysis, we consider to employ blockchain to enhance the reliability of data sharing. In addition, federated learning technology is used to deal with unreliable data owners and data requestors in secure data sharing. The main contributions of this paper are listed as follows:

(1) To address the issue of privacy protection in data sharing, we propose a data sharing mechanism based on mutual-supervised federated learning. This mechanism introduces a gradient descent algorithm based on differential privacy in the process of local model training, which protects the privacy of data contributors. In order to ensure the reliability of the data provided by the data contributors, the data contributors mutually verify the locally updated models, and the verification results will be stored in the blockchain to ensure the nonrepudiation of the verification results.

(2) The experiment results show that the proposed BPCV-FL has high accuracy and security in data sharing in IoT.

The rest of this paper is organized as follows. Section Related Work introduces the related work. Section System Model and Security Model gives the system model. Section The Implementation of the BPCV-FL elaborates the implementation of the proposed BPCV-FL. Section Performance Evaluation presents the performance evaluation. Section Conclusions concludes this paper.

2. Related Work

In order to ensure the security of data sharing, scholars have carried out extensive research and proposed corresponding solutions for the secure sharing of data in the Internet of Things by combining technologies such as federated learning, deep reinforcement learning, blockchain, and differential privacy.

Federated learning uses the idea of distributed machine learning to place model training locally on each participant, providing us with a new method to protect privacy. Fang et al. [11] proposed an efficient, privacy-preserving federated learning scheme that prevents data leakage while achieving data sharing. Sattler et al. [12] proposed a new compression framework to meet the environmental requirements of federated learning, suitable for bandwidth-constrained training environments. In traditional federated learning, each participant uploads the model parameters to the server after local training is completed after receiving the model from the server. This synchronization mode will inevitably affect the overall training efficiency. The existing scheme optimizes the efficiency of traditional federated learning. Imteaj et al. [13] performed node screening by evaluating the feedback of the participants and then iteratively updating the weight of the customer. Wang et al. and Zhang et al. [14] proposed an intelligent selection-based mechanism that intelligently selects a subset of devices to participate in federated learning to maximize rewards and thus improve verification accuracy. Zhang et al. [15] also conducted research in this area. In order to improve the efficiency of model aggregation, the authors applied deep reinforcement learning to the selection process of IIoT devices and selected devices with high-accuracy models.

Nguyen et al. [16] proposed a sharing framework for medical data in the mobile cloud environment, which combined with the decentralization of blockchain provides a solution for safe and reliable data sharing in mobile cloud computing. Zhang et al. [17] proposed a privacy-preserving data sharing model DSS-PP using blockchain and authentication technology with hidden attributes. Lalouani et al. [18] designed a new lightweight protocol based on cryptography technology to achieve the purpose of data security sharing by verifying the identity of the data recipient. Yu et al. [19] proposed a traceable and revocable blockchain data security sharing scheme in the context of smart factories. In order to solve the problem of data security sharing in the open environment of drones, Feng et al. [20] proposed an efficient and secure data sharing model by applying blockchain and attribute-based encryption.

To sum up, although the existing work has made contributions in data privacy protection, there are still deficiencies in how to ensure the reliability of shared data, for which this paper proposes a reliable data sharing scheme BPCV-FL.

3. System Model and Security Model

3.1. System Model. In this paper, we design a new data sharing mechanism BPCV-FL by improving federated learning and applying blockchain technology to the data sharing process. BPCV-FL includes the following entities.

(i) Data requester: The party who needs data will publish the data sharing task on the blockchain.

(ii) Data contributor: Known as the data node, the data contributor is the party who owns the data and is willing to participate in the data sharing. It is

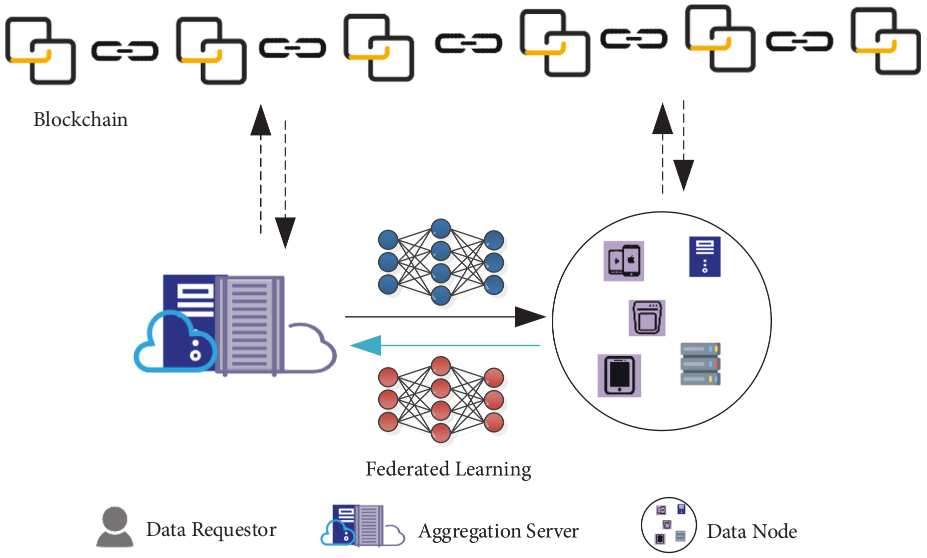

Figure 1: System model of the proposed BPCV-FL.

responsible for updating the local model and verifying the model performance of other nodes.

(iii) Aggregation server: It is responsible for aggregating the locally trained models of data contributors and distributing the aggregated global models. Different from the traditional aggregation server, in addition to distributing the global models, the aggregation server also distributes the models of other participants except each data contributor for mutual verification and supervision. The aggregation server is selected by the data requester and generated from the data node.

The system model is shown in Figure 1. As shown in the system model, the data requester publishes the request task to the blockchain, the data nodes that have relevant data and are willing to share will respond, and the data nodes participating in the data sharing process reach a consensus and get rewards. The data contributor registers on the alliance chain. The data requester sends a data sharing request task to the blockchain node connected to it. The data node that has relevant data and is willing to share responds to the task. The shared data nodes are responsible for promoting the consensus process of blockchain. The publisher of the data request task will give rewards for completing the data sharing task. The data nodes will receive corresponding rewards according to their contributions in each round of federal learning, and the data node with the worst verification result will be eliminated in each round until it meets the requirements of model accuracy or duration. After the global model training is completed, the model will be packaged and recorded on the blockchain.

The shared original data are transformed into a shared data model, and the gradient descent with differential privacy is implemented to protect the data privacy of the data provider. Data contributors validate the model with each other. The validation results are used to judge the data quality shared by data contributors, and the validation results are recorded on the blockchain [10] to ensure the reliability of shared data.

3.2. Security Model. In this paper, we consider the risks of privacy leakage and malicious damage to the model in data sharing. For the risk of privacy leakage, we consider that the data requester is interested in the private information of the object of the data being collected, and there is a possibility of exposing their privacy. For the risk of malicious damage to the model, we consider the participants in federated learning to provide malicious models, reducing the reliability of the global model. Adding differential privacy protection to local model training helps reduce the risk of privacy leakage. By introducing a mutual supervision mechanism, the reliability of the model can be guaranteed.

4. The Implementation of the BPCV-FL

The proposed strategy BPCV-FL consists of two modules, namely, the mutually supervised federated learning module and the secure data sharing module.

4.1. Mutually Supervised Federated Learning. The traditional federated learning process is that each data node uploads the model parameters to the aggregation server after completing the model training locally. The aggregation server averages the received parameters and then sends the aggregated model to the data nodes. This process iterates until the set number of rounds or time is reached. In this chapter, we optimized the traditional federated learning. The data nodes participating in the model training need to mutually verify the model accuracy uploaded to the aggregation server; that is, the data node will not only receive the aggregation model gradient issued by the aggregation server, but also have the local training model gradient of other data nodes except this node, and each data node verifies the received gradient and verifies that the data set is the local data of each data node.

Similarly, the data nodes upload to the aggregation server not only the local training model gradients of this round, but also the verification results of gradients of other data nodes.

In order to prevent dishonest participants from launching inference attacks, we introduce differential privacy into the local data of the data contributors; that is, the data contributors need to fuzzy the model parameters θ^i_{t+1}, and we consider adding disturbance to the gradient descent process. On the one hand, differential privacy can preserve the availability of model parameters; on the other hand, it can prevent inference attacks. In this chapter, we adopt the gradient descent algorithm (GD-DP) based on differential privacy. The noise added is determined by each data contributor. In order to ensure that the training process of the global model will not be eliminated by the aggregation server, the accuracy of model verification should be improved as much as possible on the premise of ensuring privacy security. The specific steps of GD-DP algorithm are as follows:

(i) For each sample L_i, we calculate

$$g(L_i) \leftarrow \nabla_\theta L(\theta, L_i). \tag{1}$$

(ii) *Step 2*: Clip $g(L_i)$ by changing the gradient vector to $g/\max(1, \|g\|_2/C)$, and the result after the limitation is either g itself or a constant C.

(iii) *Step 3*: Add noise to the gradient as follows. Define a random algorithm G, where O is any subset of the set composed of all possible outputs of G. For the sum of two adjacent data sets D and D' with at most one different record, the privacy budget of O is defined as

$$\varepsilon \triangleq \log \frac{\Pr[M(D) \in O]}{\Pr[M(D') \in O]}. \tag{2}$$

The privacy budget mainly depends on the noise scale added in the algorithm. We add Laplacian noise to the gradient

$$\widetilde{g}_i = g_i + \text{Lap}\left(\frac{\Delta f}{\varepsilon}\right), \tag{3}$$

where ε is the sensitivity defined as $\varepsilon = \max_{D,D'} \|G(D) - G(D')\|$.

(iv) *Step 4*: Perform gradient descent by

$$\theta^i_{t+1} \leftarrow \overline{\theta}_t - \eta \widetilde{g}_i, \tag{4}$$

where η is the learning rate of gradient descent when training the local model for data nodes. After adding disturbance to the θ^i_{t+1}, upload the model parameters added with disturbance to the server, and the server performs model average operation. The calculation is as follows:

$$\overline{\theta}_{t+1} \leftarrow \sum_{i=1}^{n} \left(\frac{n_i}{n}\right) \theta^i_{t+1}, \tag{5}$$

where n is the number of data nodes and n_i is the size of local data set of data node i.

After the aggregation server completes the aggregation of model parameters, the parameters distributed to each data node are divided into two parts: one is the aggregated model parameters $\overline{\theta}_{t+1}$, and the other is the model parameters of other nodes except the data node itself. After receiving the w, the data node will use the local data as the validation data set to validate each model, and upload the validation results and the locally updated model parameters to the aggregation server, and the validation results will be recorded to the blockchain as a transaction to ensure nonrepudiation. After receiving the verification results from the data nodes, the aggregation server will rank the model quality (performance) of each node, eliminate the worst model performance, and then add a data node with the most timely response according to the response time. For data node i, we record the verification results of other nodes except for i itself as that the aggregation server calculates the model quality performance of the data node with

$$v^i_r = v^1_r + v^2_r + \cdots + v^n_r. \tag{6}$$

For the identification task, v^i_r is the sum of the accuracy verified for each data node and v^i_r is the sum of the average absolute errors for the recursive task. Considering that the data node may temporarily exit the model training process due to dissatisfaction with the reward, in order to avoid this situation, the task publisher needs to publish the reward size of each round. The request task issued by the data requester consists of three parts: one is the request category, such as the identification of an object, the second is the number of rounds or maximum time limit of the federal learning and training model, and the third is the reward for completing the task. In addition to the aggregation server disconnecting the data node with the worst verification result, the data node can also decide whether to respond to the task according to the reward budget.

In each round of model training, the aggregation server ranks the model validation results of each data node after receiving the validation results. On the one hand, the server will eliminate the last data node according to the verification results to improve the reliability and efficiency of model training.

4.2. Secure Data Sharing Procedure.

The data requester publishes the request task on the blockchain, and each task consists of three parts, namely, the requested data category, the number of rounds or time threshold of federated learning iteration, and the reward of each round of federated learning. After the data requester publishes the task, the data nodes that have the relevant data and are willing to share it will respond to the task, and one of the nodes will be selected by the task publisher as the aggregation server. The task of the aggregation server is to initialize the model, aggregate the models uploaded by each data node for the next round of federated learning, and then perform model training according to the proposed mutual-supervised federated learning algorithm. During the training of the data model, the results of each round of mutual verification between data nodes will be broadcast as blockchain transactions, and these transactions will be packaged into blocks by the accounting node [21]. The consensus process is carried out among the data nodes participating in data sharing. The data nodes compete for the right of accounting according to the quality of the local model, that is, the opportunity to package transactions into blocks. The data node that has obtained the accounting right will broadcast the generated block to other nodes for verification. After the verification is passed, the block will be added to the blockchain.

In order to solve the privacy and security issues of data owners in the process of data sharing, and at the same time improve the quality of shared data, we consider the combination of blockchain and federated learning. Through mutual supervision of data nodes and recording the verification results on the blockchain, security auditing becomes possible. Considering that consensus based on workload mechanism consumes a lot of resources, this paper adopts a consensus mechanism based on model quality (PoQ). PoQ can reach consensus among data nodes according to the verification results of the model, which not only saves computing resources, but also improves the efficiency of reaching consensus. The nodes responsible for packing shared transactions into blocks are determined based on the quality of the model training. Traditional federated learning cannot effectively supervise data nodes, because data nodes may have some local models, which affect the quality or training efficiency of the global model. To avoid this problem, each data node in the optimized federated learning needs to verify the models of other nodes in addition to updating the model. The verification result is used as the evaluation criterion for the quality of the locally updated model for each data node.

(i) For classification tasks, the local update model quality of each node can be measured as the accuracy of model validation, which is defined by

$$\mathrm{Acc} = \frac{T_p}{T_p + F_p}, \tag{7}$$

where T_p is the number of positive samples correctly detected and F_p is the number of positive samples incorrectly detected.

(ii) For regression tasks, model quality can be measured as the absolute mean error of model validation, that is, the average distance between the predicted value of the model and the real value of the sample as

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N} \left|\mathrm{Act}(i) - f(i)\right|, \tag{8}$$

where N is the total number of samples, $\mathrm{Act}(i)$ is the real value of sample i, and $f(i)$ is the predicted value of sample i.

When the consensus is implemented, the data nodes participating in the consensus vote to select the node with the highest training accuracy or the lowest MAE of the entire federal learning model as the accounting node. The accounting node is responsible for packaging the data sharing transactions into blocks. After the newly generated block is broadcast to other data nodes, each node verifies the block and recorded transactions. If the verification passes, the block will be written to the blockchain. Another advantage of the consensus mechanism of our proposed scheme is that it can exclude data nodes with low data quality and promote reliable data sharing among nodes.

4.3. Security Analysis. The scheme proposed in this paper is based on mutual verifiable federated learning and blockchain. The tamperability of blockchain provides certain security for this scheme.

(i) Security of Verification Results: Since the model performance results of each round of verification of each data node are recorded on the alliance chain, they are invisible to nodes that do not belong to the alliance chain and cannot be tampered with, which ensures the security of the verification results to a certain extent.

(ii) Data Privacy Security: The gradient descent algorithm with differential privacy is implemented. Within the acceptable range of impact on the accuracy of the model, adding a certain amount of noise to the model parameters has a good effect on privacy protection. As the risk of privacy disclosure increases, the algorithm performs better in terms of privacy protection than FedPAGE [22] and Overlap-FedAvg [23].

5. Performance Evaluation

5.1. Experimental Environment. The experiment is conducted on a computer equipped with a Windows 7 system. The machine was equipped with an Intel Core i7 processor with 6.4 GHZ CPU frequency. The Python programming language was used to verify the effectiveness of the proposed strategy. First, we verify the effectiveness of the proposed mutually supervised federated learning, and then we carry out experiments on the performance of the blockchain. We evaluated on the MNIST data set and the CIFAR-10 data set. MNIST data set is a data set of handwritten digital pictures, which contains 60000 images and labels of 10 digital categories from 0 to 9, and the size of each picture is 28 ∗ 28. The CIFAR-10 data set contains 60000 color pictures of 10 categories, such as aircraft, cars, and birds. There are 6000 pictures in each category, and the size of each picture is 32 ∗ 32. These two data sets are widely used in the evaluation of classification tasks. We randomly segment MNIST and CIFAR-10 data sets to simulate the characteristics of small-scale data owned by various institutions or individuals in the actual situation. For the blockchain configuration, we set that the numbers of data nodes are 30 and 50, the blockchain transaction volume is set to five transactions per second, and the block generation rate is set to 1 pieces by two seconds.

In order to verify the feasibility of the proposed mechanism BPCV-FL, we conduct experimental analysis on the following two scenarios: a federated learning scenario with 30 data nodes and a federated learning scenario with 50 data nodes. The specific experimental indicators are as follows:

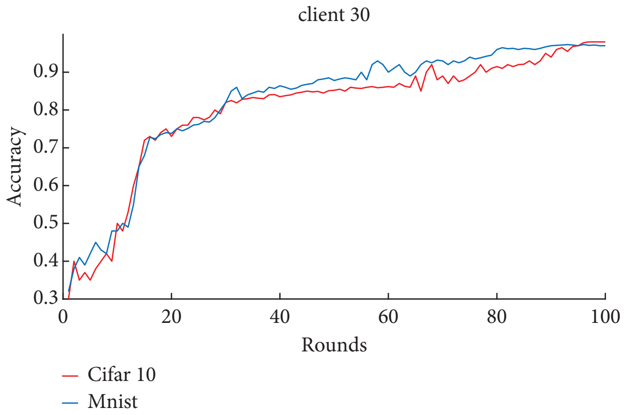

Figure 2: Accuracy of 30 clients.

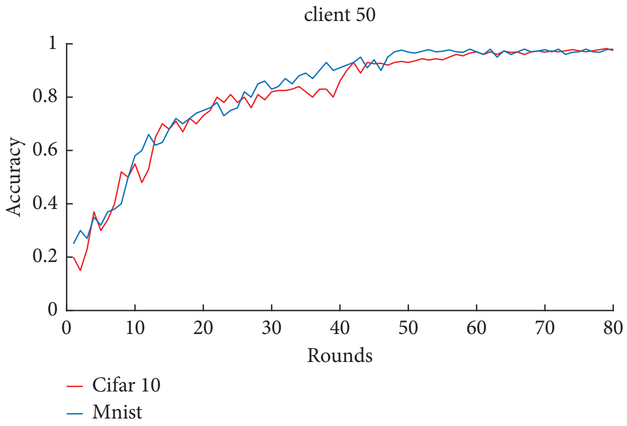

Figure 3: Accuracy of 50 clients.

(i) Accuracy Rate: It refers to the average value of the accuracy rate verified by the global model in the local data set of each data contributor.

(ii) Privacy Protection Degree: It refers to the protection degree of GP-DP algorithm to the privacy of the global model.

(iii) CPU Utilization: It refers to the CPU occupancy of the client when running the blockchain system.

6. Experimental Results

To verify the effectiveness of the proposed mutual-supervised federated learning, we verify the accuracy of training the global model for scenarios with 30 data nodes and 50 data nodes, respectively. As can be seen from Figure 2, in the scenario of 30 clients, for the data set CIFAR-10, the global model needs about 90 rounds of convergence, and for the data set MNIST, the global model needs about 80 rounds of convergence. Finally, the accuracy of the two global models can reach about 95%, which shows that the scheme proposed in this chapter performs well in effectively ensuring the high accuracy of the global model.

As shown in Figure 3, with 50 clients, it takes about 60 epochs for the CIFAR-10 data set to converge the global model and about 45 epochs for the MNIST data set to converge. The more the clients, the faster the convergence, because more data features are covered.

In order to verify the effect of the gradient descent algorithm with differential privacy on the global model accuracy, we compared the changes in model training accuracy with and without perturbation for the above 30 data nodes and 50 data nodes, respectively. The data set used is MNIST, as shown in Figures 4 and 5. We set the local data privacy budget of data nodes to 10. As can be seen from Figures 4 and 5, adding noise does not cause much loss to the accuracy of the model.

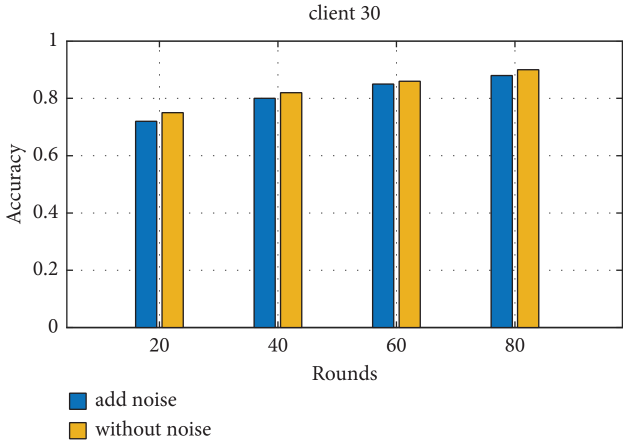

Figure 4: Accuracy of executing GD-DP algorithm with 30 clients.

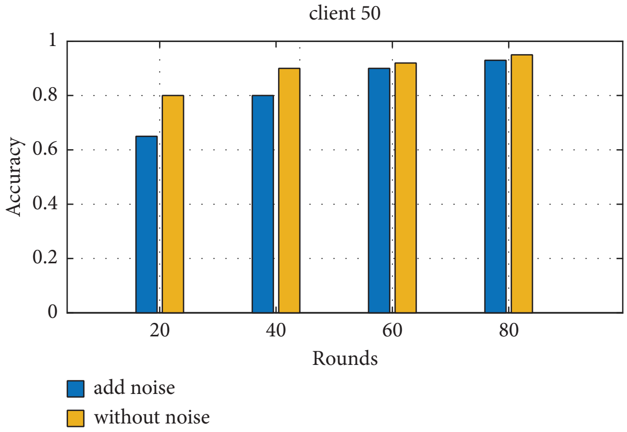

Figure 5: Accuracy of executing GD-DP algorithm with 50 clients.

As shown in Figure 4, in the case of 30 clients, the accuracy of federated learning is 0.72 with scrambling and 0.75 without scrambling for 20 rounds. When federated learning runs for 40 rounds, the scrambled accuracy is 0.8, and the unscrambled accuracy is 0.82. When the model converges, the scrambled accuracy is 0.95, and the unscrambled accuracy is 0.97.

Similarly, in Figure 5, under the same number of training rounds, the accuracy of scrambling is only less than 4% lower than that without scrambling. This difference is acceptable because adding interference will definitely better protect the local data privacy of data nodes. Therefore, it can be seen that

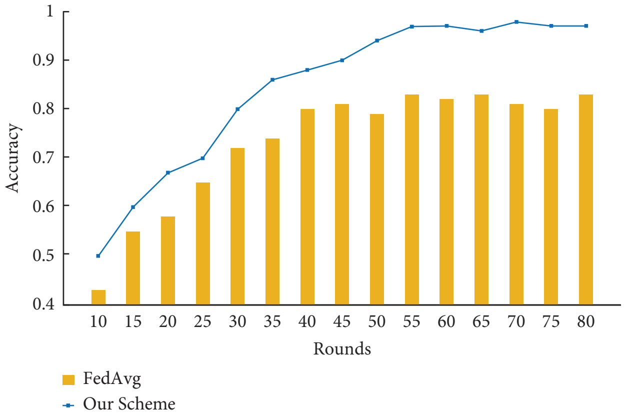

Figure 6: Comparison of accuracy under 50 clients.

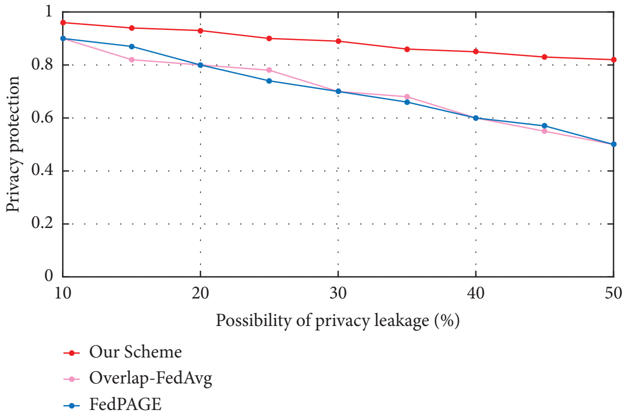

Figure 7: Privacy comparison.

our proposed gradient descent with differential privacy is feasible. According to the above experiments, we can see that the global model of the proposed federated learning algorithm achieves a high level of accuracy when the number of data nodes is different, which shows that the implementation of the gradient descent algorithm with differential privacy will not be correct. The accuracy of the global model has a large impact, which demonstrates the effectiveness of our proposed scheme.

Furthermore, we compare the performance of the proposed algorithm and the FedAvg aggregation algorithm in the case of malicious behavior, as shown in Figure 6. We set up a total of 50 data nodes participating in federated learning, including 5 malicious nodes. From the figure, we can see that the algorithm proposed in this paper performs better in the case of malicious behavior, because in the algorithm proposed in this paper, the model update results will be verified in each round of federated learning. Therefore, it can be seen that the algorithm proposed in this paper has a higher guarantee on the accuracy of the model.

We compared the performance of gradient descent algorithm with differential privacy and FedPAGE [22] and Overlap-FedAvg [23] algorithm in terms of privacy protection, and the results are shown in Figure 7. In this figure, there are 30 clients and each client has a local privacy budget of 10. The GD-DP algorithm proposed in this paper is better than the FedPAGE algorithm and the Overlap-FedAvg algorithm, which proves that the scheme has better privacy protection.

What we store on the blockchain is the result index of the model verification of other nodes by nodes participating in federated learning. In order to test its performance, we built an Ethereum private chain, and by writing related smart contracts, the nodes that contributed the most to package transactions into blocks. In addition, we set the transaction volume to 5 transactions per second and the block generation rate to generate a block every two seconds, and write the system performance data to the time series database to record the performance data that changes over time. The experimental results are shown in Figures 8 and 9 in terms of the CPU utilization of the system with 30 and 50 clients, respectively. The geth node is one of the last welcomed clients in Ethereum. It can be seen that the usage rate of the client CPU is low, which has a good effect on saving the local computing resources of the data node.

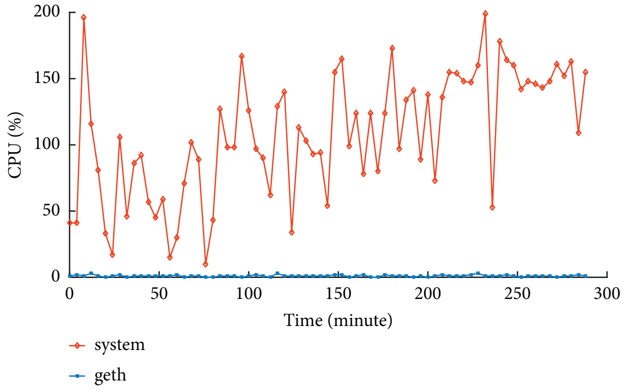

Figure 8: System CPU utilization under 30 clients.

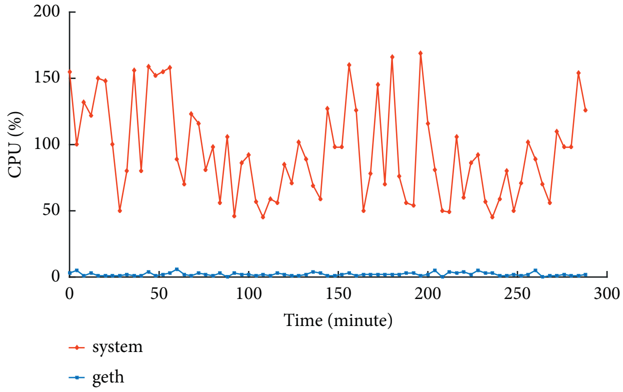

Figure 9: System CPU utilization under 50 clients.

7. Conclusions

In this paper, we propose a mutually supervised data sharing mechanism based on mutual-supervised federated learning. The shared source data are transformed into a shared data model to protect the data privacy of the data contributors participating in the sharing. In the process of local model training, a gradient descent algorithm based on differential privacy is introduced to further protect the privacy security of the data contributors. In order to ensure the reliability of the data provided by the data contributors, the data contributors supervise each other and mutually verify their locally updated models. The quality of the verification results will be stored in the blockchain to ensure the non-repudiation. Simulation results show that the proposed scheme is effective and has better performance in protecting data privacy and improving data reliability.

Data Availability

The minist dataset and Cifar 10 dataset are used to support the findings of this study, which are available at "https://yann.lecun.com/exdb/mnist/" and "http://www.cs.toronto.edu/~kriz/cifar.html," respectively.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by National Natural Science Foundation of China, under Grant nos. U1905211 and 61702103, and Natural Science Foundation of Fujian Province, under Grant nos. 2022J01644, 2020J01167, and 2020J01169.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: a survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[2] K. J. Singh and D. S. Kapoor, "Create your own internet of things: a survey of iot platforms," *IEEE Consumer Electronics Magazine*, vol. 6, no. 2, pp. 57–68, 2017.

[3] J. King and I. A. Ali, "A distributed security mechanism for resource-constrained iot devices," *Informatica*, vol. 40, no. 1, 2016.

[4] M. Bali, A. Tari, A. Almutawakel, and O. Kazar, "Smart design for resources allocation in iot application service based on multi-agent system and csp," *Informatica*, vol. 44, no. 3, 2020.

[5] A. Whitmore, A. Agarwal, and L. Da Xu, "The internet of things-a survey of topics and trends," *Information Systems Frontiers*, vol. 17, no. 2, pp. 261–274, 2015.

[6] H. A. Tran, D. Tran, L. G. Nguyen, Q. T. Ha, V. Tong, and A. Mellouk, "Shiot: a novel sdn-based framework for the heterogeneous internet of things," *Informatica*, vol. 42, no. 3, 2018.

[7] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: challenges, methods, and future directions," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 50–60, 2020.

[8] Q. Miao, H. Lin, X. Wang, and M. M. Hassan, "Federated deep reinforcement learning based secure data sharing for internet of things," *Computer Networks*, vol. 197, Article ID 108327, 2021.

[9] J. Mills, J. Hu, and G. Min, "Communication-efficient federated learning for wireless edge intelligence in iot," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 5986–5994, 2020.

[10] Y. Xu, C. Zhang, Q. Zeng, G. Wang, Ju Ren, and Y. Zhang, "Blockchain-enabled accountability mechanism against information leakage in vertical industry services," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 2, pp. 1202–1213, 2021.

[11] C. Fang, Y. Guo, N. Wang, and A. Ju, "Highly efficient federated learning with strong privacy preservation in cloud computing," *Computers & Security*, vol. 96, Article ID 101889, 2020.

[12] F. Sattler, S. Wiedemann, K. R. Muller, and W. Samek, "Robust and communication-efficient federated learning from non-iid data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 9, pp. 3400–3413, 2020.

[13] I. Ahmed and M. H. Amini, "Distributed sensing using smart end-user devices: pathway to federated learning for autonomous iot," in *Proceedings of the 2019 International conference on computational science and computational intelligence (CSCI)*, pp. 1156–1161, IEEE, Las Vegas, NV, USA, December 2019.

[14] H. Wang, Z. Kaplan, Di Niu, and B. Li, "Optimizing federated learning on non-iid data with reinforcement learning," in *Proceedings of the IEEE INFOCOM 2020-IEEE Conference on Computer Communications*, pp. 1698–1707, IEEE, Toronto, ON, Canada, July 2020.

[15] P. Zhang, C. Wang, C. Jiang, and Z. Han, "Deep reinforcement learning assisted federated learning algorithm for data management of iiot," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 12, pp. 8475–8484, 2021.

[16] D. C. Nguyen, P. N. Pathirana, M. Ding, and A. Seneviratne, "Blockchain for secure ehrs sharing of mobile cloud based e-health systems," *IEEE Access*, vol. 7, Article ID 66792, 2019.

[17] Q. Zhang, Y. Li, R. Wang, Lu Liu, Yu-an Tan, and J. Hu, "Data security sharing model based on privacy protection for blockchain-enabled industrial internet of things," *International Journal of Intelligent Systems*, vol. 36, no. 1, pp. 94–111, 2021.

[18] W. Lalouani, M. Younis, M. Ebrahimabadi, and N. Karimi, "Robust and efficient data security solution for pervasive data sharing in iot," in *Proceedings of the 2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC)*, pp. 775–781, IEEE, Las Vegas, NV, USA, January 2022.

[19] K. Yu, L. Tan, M. Aloqaily, H. Yang, and Y. Jararweh, "Blockchain-enhanced data sharing with traceable and direct revocation in iiot," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 11, pp. 7669–7678, 2021.

[20] C. Feng, K. Yu, A. K. Bashir et al., "Efficient and secure data sharing for 5g flying drones: a blockchain-enabled approach," *IEEE Network*, vol. 35, no. 1, pp. 130–137, 2021.

[21] Y. Xu, C. Zhang, G. Wang, Z. Qin, and Q. Zeng, "A blockchain-enabled deduplicatable data auditing mechanism for network storage services," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 3, pp. 1421–1432, 2021.

[22] H. Zhao, Z. Li, and P. Richtárik, "Fedpage: A Fast Local Stochastic Gradient Method for Communication-Efficient Federated Learning," 2021, https://arxiv.org/abs/2108.04755.

[23] Y. Zhou, Q. Ye, and J. Lv, "Communication-efficient federated learning with compensated overlap-fedavg," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 1, pp. 192–205, 2022.