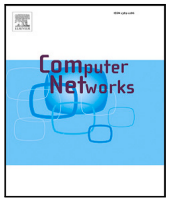
Computer Networks 252 (2024) 110691

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet



TPE-BFL: Training Parameter Encryption scheme for Blockchain based Federated Learning system

Fanfan Shen [a], Qiwei Liang [a,*], Lijie Hui [a], Bofan Yang [a], Chao Xu [a], Jun Feng [b], Yanxiang He [c]

[a] School of Computer Science, Nanjing Audit University, Nanjing 211815, China
[b] School of Cyber Science and Engineering, Huazhong University of Science and Technology, Wuhan, Hubei 430074, China
[c] School of Computer Science, Wuhan University, Wuhan 430072, China

ARTICLE INFO

Keywords:
Federated Learning
Blockchain
Paillier
Training parameter encryption
Privacy protection

ABSTRACT

Blockchain technology plays a pivotal role in addressing the single point of failure issues in federated learning systems, due to the immutable nature and decentralized architecture. However, traditional blockchain-based federated learning systems still face privacy and security challenges when transmitting training model parameters to individual nodes. Malicious nodes within the system can exploit this process to steal parameters and extract sensitive information, leading to data leakage. To address this problem, we propose a Training Parameter Encryption scheme for Blockchain based Federated Learning system (TPE-BFL). In TPE-BFL, the training parameters of the system model are encrypted using the paillier algorithm with the property of addition homomorphism. This encryption mechanism is integrated into the workflows of three distinct roles within the system: workers, validators, and miners. (1) Workers utilize the paillier encryption algorithm to encrypt training parameters for local training models. (2) Validators decrypt received encrypted training parameters using private keys to verify their validity. (3) Miners receive cryptographic training parameters from validators, validate them, and generate blocks for subsequent global model updates. By implementing the TPE-BFL mechanism, we not only preserve the immutability and decentralization advantages of blockchain technology but also significantly enhance the privacy protection capabilities during data transmission in federated learning systems. In order to verify the security of TPE-BFL, we leverage the semantic security inherent in the Paillier encryption algorithm to theoretically substantiate the security of our system. In addition, we conducted a large number of experiments on real-world data to prove the validity of our proposed TPE-BFL, and when 15% of malicious devices are present, TPE-BFL achieve 92% model accuracy, a 5% improvement over the blockchain-based decentralized FL framework (VBFL).

1. Introduction

In the contemporary era, the proliferation of cutting-edge technologies, including big data and blockchain, has significantly enhanced the convenience of everyday life for individuals. However, it has also led to issues concerning the privacy and security of user's data. In response to data privacy issues, governments around the world have developed strategic frameworks aimed at protecting personal information. The European Union's General Data Protection Regulation (GDPR) is a prime example, setting a precedent for comprehensive data protection. At the same time, China has also introduced a cybersecurity Law, further demonstrating the international commitment of countries to strengthen digital privacy protection through legislative means. These initiatives highlight the determination of countries to work together to build a secure data environment, reflecting the importance of privacy in the digital age [1–3]. The introduction of these plans has imposed many restrictions on privacy data, resulting in an increasing amount of data from various fields being scattered across different organizations, which has led to the phenomenon of data silos. In light of these problems, Federated Learning (FL), as a distributed training method, is undoubtedly an excellent solution. FL keeps data local by facilitating collaborative model training that eliminates the need for centralized aggregation of user data. In addition, FL with privacy protection features can effectively mitigate the risk of personal sensitive data breaches to a considerable extent [2–4].

Despite its advantages in safeguarding data privacy, FL faces challenges, notably the risk of a single point of failure due to its reliance on a central server. Should this server fall prey to malicious attacks, it could cause systemic collapse, endangering users' sensitive data and

* Corresponding author.
E-mail addresses: ffshen@whu.edu.cn (F. Shen), MP2209117@stu.nau.edu.cn (Q. Liang), MP2209103@stu.nau.edu.cn (L. Hui), MP2209118@stu.nau.edu.cn (B. Yang), xuchao@nau.edu.cn (C. Xu), junfeng@hust.edu.cn (J. Feng), yxhe@whu.edu.cn (Y. He).

https://doi.org/10.1016/j.comnet.2024.110691
Received 11 May 2024; Received in revised form 17 July 2024; Accepted 1 August 2024
Available online 5 August 2024

compromising its integrity and confidentiality [5,6]. To counteract this vulnerability, integrating blockchain technology has been proposed. Blockchain technology, with its immutability, security, and smart contract capabilities, provides a secure and shared data storage solution through a distributed network and consensus mechanism. The integration with FL lessens the central server's dominance, bolstering the system's security and reliability. Moreover, the system's vulnerability caused by the single point of failure is substantially reduced [5,7–9].

Currently, many scholars have conducted researches on blockchain-based federated learning systems [10–12]. The application of blockchain technology not only effectively avoids the single point of failure issue in FL but also enhances the system's security. Issa et al. [10] conducted an in-depth analysis of the security protocols within a blockchain-augmented federated learning framework, specifically tailored for the Internet of Things (IoT). BC-FL improves the security of data and systems by leveraging the decentralized and tamper-resistant nature of blockchain technology. Additionally, the application of blockchain technology endows IoT systems with verifiability, encouraging honest operation from each device. In [11], the authors utilized blockchain technology to connect servers, clients, and aggregators within the FL framework, recording the information flow involving federated learning operations, participating clients, and updates to both local and global models on a distributed ledger of transactions. This approach prevents single point of failure attacks and strengthens the system's resilience against attacks. In [9], a blockchain-based decentralized verification mechanism was proposed that, by defining roles, limits the power and responsibilities of each, thereby reducing the adverse impact of malicious nodes in blockchain federated learning systems. This mechanism enhances system security while preventing single point of failure attacks.

However, FL architectures intergrate with Block-chain (BFL) are also susceptible to adversarial and inference attacks, these attacks can result in the disclosure of private data [10,12,13]. Due to the transparent nature of blockchain, data stored on the blockchain is openly accessible to all participants within the network. That is, trained model parameters are stored and broadcast to all devices. When there are curious or malicious participants, they can easily infer each other's information based on parameters and other raw data, leading to data leakage [12,13]. Moreover, the approach of FL, which stores data independently on local client devices for training, does not fully guarantee data privacy and security. Training datasets may be obtained by servers from shared gradients [14] public nature of model updates makes them susceptible to poisoning attacks [15].

To tackle ongoing privacy and security issues in Blockchain-based Federated Learning (BFL) systems, researchers have suggested using cryptographic methods to prevent data exposure [11,16–18]. Shayan et al. [16] introduced Biscotti, a BFL system that emphasizes privacy and security. This system leveraged the peer-to-peer architecture of blockchain to minimize dependence on centralized servers. It integrated differential privacy by introducing random noise into model updates, ensuring secure client data against unauthorized access. Bin et al. [17] combined differential privacy with an advanced Paillier homomorphic encryption algorithm in a model that merges federated learning with blockchain's distributed framework, offering robust protection for data privacy and security that ensures robust security within the Industrial IoT devices. Despite these advancements, which significantly enhance system security, the exchange of training parameters between devices within the system continues to present privacy risks.

To tackle the aforementioned issues, we propose a Training Parameter Encryption scheme for Blockchain based Federated Learning system (TPE-BFL). TPE-BFL employs the Paillier algorithm to ensure the encrypted transmission of training parameters throughout the duration of the training phase. Given the homomorphic properties of the Paillier algorithm, it allows for operations on encrypted training parameters without the need for decryption, thereby reducing the risk of privacy leakage during transmission. Theoretically, we have demonstrated the security of the proposed TPE-BFL. Extensive experiments conducted on both the MNIST and Fashion-MNIST datasets demonstrate that our approach outperforms several benchmark methods. Our method (TPE-BFL) maintains an optimal model accuracy rate even when the proportion of malicious nodes is at 15% and increases to 20%.

To the end, we summarize our main contributions as:

- We propose a privacy protection framework for Blockchain-based Federated Learning that aims to secure the transmission of training parameters by encrypting them, ensuring their confidentiality and protection against unauthorized disclosure.
- We propose the Training Parameter Encryption scheme for Blockchain-based Federated Learning (TPE-BFL), utilizing the paillier homomorphic encryption algorithm to secure the transmission of training parameters among three defined roles within the system. To the best of our knowledge, this is the first time to encrypt training parameters in a role-segregated blockchain-federated learning context, significantly reducing the risk of parameter exposure. Furthermore, we provide a theoretical proof of the robust security measures inherent in TPE-BFL.
- We conducted a number of rigorous experiments on two datasets, MNIST and Fashion-MNIST, to verify the efficiency of TPE-BFL. The experimental results indicate that TPE-BFL achieves higher test accuracy in the event of a rise in malicious nodes.

The organization of the remaining part of this paper is as follows. We review the related work in Section 2 and the methodology are introduced in Section 3. We prove the safety of TPE-BFL by theoretical analysis in Section 4. Section 5 describes the datasets and the experimental analysis. Finally, we conclude this article in Section 6.

2. Related work

In this section, we will introduce the related background knowledge about blockchain based FL and paillier algorithm.

Blockchain based FL: The progressive highlighting of privacy conservation has provoked to increased scrutiny of traditional training methods, which involve conveying the original data set to a server and consequently heighten the risk of privacy breaches. In 2016, McMahan et al. [19] initinated the theory of "federated learning" in response to this privacy risk, wherein participant data is maintained in local storage and the shared model is developed through the aggregation of individual computation updates. FL protects users' privacy to some extent, Meng et al. [20] successfully applied FL architecture to the field of industrial artificial intelligence (AI) and proposed a privacy-enhanced Federated Learning architecture (PEFL). This architecture used a non-interactive security aggregation method combined with distributed Gaussian mechanism to achieve differential privacy protection at sample level. This innovation provided a solid guarantee for data privacy and security in the industrial field, while promoting the application and development of AI technology in industrial environments. Konecn et al. [21] proposed a FL framework using structured updates and sketched updates to both cut down on transmission overhead and meet the requirements of data privacy protection. Although these methods overcame the challenge of data privacy security to a certain point, the traditional federated learning model they use is server-client architecture, which depends on one server to carry out updates to the global model. When the central server is subjected to a malicious target, it will seriously affect the entire training process, and the original data of the client will also face the risk of leakage.

In response to the single point of failure crisis in Federated Learning (FL), some scholars have proposed the application of blockchain technology to achieve decentralized federated learning. By integrating blockchain technology with federated learning, Chuan et al. [7] have introduced a novel distributed framework known as BLADE-FL, that achieved complete decentralization by establishing a P2P network between task publishers and training clients. Korkmaz et al. [22] applied

Ethereum to a FL model and proposed the Chain FL framework which implements a distributed data storage platform through Ethereum, allowing model data to be stored on multiple nodes without relying on a single central server. In addition, POA consensus algorithm is used to select reputable verifiers for block creation and transaction verification, thereby improving the transaction processing speed and enhancing the robustness of the model [22]. The author proposed the BC-FL framework in the paper [23], which utilized blockchain technology to store model and summarized the update of the model through smart contracts, avoiding the single point of failure caused by the excessive power of the central server of traditional federated learning. However, the method proposed necessitates substantial data transmission among devices during model updating and training, resulting in increased communication and computational burdens.

Fan et al. [12] proposed a model LPBFL, which can realize lightweight privacy and security computation, and improved the efficiency of verification by designing lightweight digital signature and batch verification algorithm. In addition, the author's proposed LPBFL incorporates a hierarchical framework and employs a homomorphic encryption algorithm. This approach guarantees both model accuracy and privacy while simultaneously minimizing computational and communication burdens. However, blockchain based FL framework proposed by the above research lacks the setting of incentive mechanism, which affects the enthusiasm of participants in the model training process to a certain extent. By setting an incentive mechanism to provide rewards to each participant, the model's training effectiveness has been enhanced, and it has been made to enhance the system's security and resilience. In the paper [9], the researchers introduced VBFL, a decentralized framework for Federated Learning (FL) based on blockchain technology. This framework employs a single verifier to authenticate model updates, ensuring their validity. Additionally, the authors implemented an incentive mechanism using the POS consensus algorithm to enhance the efficiency and security of model training while rewarding trustworthy devices.

However, during the training of the model, there is still a possibility of local model parameter leakage, which can lead to the model's inability to handle malicious actor attacks. This can result in decreased overall model accuracy and compromised model robustness. In order to solve the above problems, paillier homomorphic encryption algorithm is undoubtedly an effective solution, and the security of the system can be guaranteed through the encryption operation of the model parameters.

Paillier Algorithm: paillier homomorphic encryption algorithm with asymmetric encryption characteristics is the most effective addition homomorphic encryption system at present, which provides semantic security against selected plaintext attacks (IND-CPA) [24,25]. Due to the homomorphism of paillier algorithm, encrypted data can be processed without exposing the original data [26]. It provides an important guarantee for data privacy and security [27]. Because this algorithm has good encryption ability, many scholars have applied paillier homomorphic encryption algorithm to federated learning to further improve privacy security. Through the utilization of encryption techniques on both model parameters and gradients during the training process, sensitive data can be safeguarded against unauthorized access. Additionally, this approach enables collaborative machine learning without necessitating data sharing among participants, thereby ensuring efficient model training. Moreover, the implementation of this algorithm guarantees that privacy protection is enhanced while maintaining high training accuracy.

In [28–30], the authors integrated the paillier homomorphic encryption algorithm with federated learning to secure the model gradient during training by encrypting it using the paillier algorithm. This approach effectively mitigates privacy risks associated with model training. PFMLP, a privacy protection framework for multi-party federal learning proposed by Fang et al. [28], utilizes paillier homomorphic encryption to encrypt and handle the gradient data trained by the client. Even in the event of a malicious attack on the central server, only the encrypted gradient data will be exposed, effectively mitigating the risk of inference attacks. In [29], the researchers incorporated the paillier encryption algorithm into federated learning for processing the shared gradient. Additionally, they introduced bilinear aggregate signature technology to ensure the accuracy of the model and enhance privacy security by verifying the correctness of the aggregate gradient. However, the computational overhead and efficiency of model training are compromised due to the utilization of the paillier algorithm. In literature [30], the researchers proposed an innovative batch encryption technology. Before the encryption operation is implemented, the technique first quantifies the gradient values used for model training and then performs batch encoding. This approach not only optimizes the data processing process, but also enhances the security and efficiency of the encryption process, providing an effective solution for protecting sensitive data.

However, gradient encryption still has some security problems. Gradient encryption does not protect all parameters of the model. When there are malicious attackers, these malicious attackers may circumvent gradient encryption protection and access the parameter information of the model. In light of these challenges, certain academics have suggested the incorporation of encryption techniques during the training phase to safeguard model parameters [31,32]. Ma et al. [31] proposed a Byzantine federated learning privacy protection mechanism to protect model parameters transmitted by worker nodes from being leaked through distributed paillier encryption algorithm. In [32], the researchers developed a robust aggregation protocol by utilizing the paillier homomorphic encryption algorithm. By applying encryption to the model parameters shared with the central server, potential malicious activities aimed at unauthorized acquisition of these parameters are effectively prevented.

Aiming at aforementioned problems, some researchers have brought breakthrough improvements to the federated learning architecture through innovative fusion of blockchain technology and the Paillier homomorphic encryption algorithm. This combination not only solves the problem of single point of failure in traditional federated learning, but also significantly improves the security and stability of the system.

In [11], the authors proposed the blockchain-based privacy-perserving federated learning framework, an innovative scheme that combines federation learning, blockchain technology, and paillier homomorphic encryption algorithm. Using the immutability and decentralization of blockchain, it provides a solid source and verification mechanism for model update. At the same time, the gradient is encrypted by the paillier encryption algorithm, which effectively protects the local model in the process of model exchange and prevents data leakage. Nevertheless, within the BC-based PPFL framework, the server takes on the pivotal role of validator, tasked with assessing the aggregated gradients post-iteration. This design significantly amplifies the server's workload and exposes it to heightened information security risks. The integrity and operation of the entire system could be jeopardized in the event of a server attack or malfunction, concurrently escalating the potential for privacy breaches. In addition, although the BC-based PPFL framework theoretically shifts from the semi-honest client hypothesis to the malicious client hypothesis to accommodate a more complex security environment, the framework does not directly simulate or introduce malicious clients in experiments to adequately validate system performance. This means that the ability to resist malicious behavior has not yet been tested in practice.

Differences in TPE-BFL: Inspired by the above work, we propose a Training Parameter Encryption scheme for Blockchain based Federated Learning System. The paillier homomorphic encryption algorithm is utilized by us to develop a secure aggregation protocol. By employing encryption on the model parameters that are uploaded to the central server, potential malicious actions from servers attempting to steal these parameters are effectively prevented. This is particularly important as the federation learning in a server-client architecture is

susceptible to a crisis caused by a single point of failure, and even blockchain-based federation learning systems carry risks of privacy disclosure during model training.

Considering these problems, based on the research conducted by Chen et al. [9], we have implemented a technique to secure the transmission of local model training parameters throughout the communication process. To the best of our knowledge, we are pioneers in proposing an encryption method for safeguarding transmitted training parameters within a blockchain-based federated learning system that utilizes role partitioning. By introducing additional malicious nodes and gradually increasing their number, our TPE-BFL approach has demonstrated its ability to maintain high accuracy in model training while resisting these destructive behaviors. This strategy not only strengthens the system's defense against malicious attacks, but also ensures the accuracy and reliability of the learning process. In addition, we also conduct an in-depth analysis of the safety of TPE-BFL from the theoretical level. Through the combination of experiment and theory analysis, the robustness and stability of our framework in the face of complex security challenges are more comprehensively demonstrated.

3. Methodology

In this section, we will introduce some preliminaries and our novel approach, TPE-BFL. This discussion will encompass a detailed explanation of the algorithmic process and the architectural framework of the model.

A. Preliminaries and Notations

- Paillier Algorithm
  The fundamental technology employed in TPE-BFL is Paillier homomorphic encryption. We utilize the paillier encryption algorithm to secure local model parameters, enabling multiple entities to engage in the process of encrypting and decrypting. The specifics are outlined below:
  (1) Key Generation: Choose two similar large prime numbers p and q, and calculate N = pq, where the number of bits of N determines the length of the key, then calculate the Euler function.

$$\varphi(N) = (p-1)(q-1) \tag{1}$$

$$\gcd(pq, (p-1)(q-1)) = \gcd(pq, \varphi(N)) = 1 \tag{2}$$

  Select a generator g $g \in Z^*_{N^2}$ that is the primary root of module N; Select a random number λ and calculate a modular inverse L that satisfies

$$\lambda = lcm(p-1, q-1) \tag{3}$$

$$L * \lambda \equiv 1 \bmod \varphi(N) \tag{4}$$

  where the public key is $Pubkey = (N, g)$, the private key is Pr $ivkey = \lambda$.
  (2) Encryption: Let the plaintext be m, where $m \in Z_N$, a random number r is selected, and the ciphertext is calculated

$$C = g^m * r^s \bmod N^2 \tag{5}$$

  where s is another random number.
  (3) Decryption: After the key is given, the private key is used to decrypt and calculate the plaintext

$$m = L(c^\lambda \bmod N^2) \cdot \mu \bmod N \tag{6}$$

  where μ is a modular inverse of N, used for decryption operations.

$$\mu = (L(g^\lambda \bmod N^2))^{-1} \tag{7}$$

  We use paillier encryption algorithm in the proposed TPE-BFL. TPE-BFL satisfies the characteristics of addition and multiplication homomorphic encryption, and can directly perform addition and multiplication operations without decrypting data first.

- Notations
  To clarify our approach, we describe the symbolic representations used in Table 1.

B. The proposed TPE-BFL system

In this section, we present a novel approach called Training Parameter Encryption scheme for Blockchain-based Federated Learning system (TPE-BFL). TPE-BFL utilizes the paillier encryption algorithm to secure the transmission of training parameters among the three roles involved in the system. To our knowledge, this is the first proposed method for encrypting training parameters within a role-segregated blockchain-based federated learning system. In contrast to the conventional federated learning framework that relies on a central server, the TPE-BFL system presented in this paper introduces a tripartite division of roles, each imbued with distinct responsibilities. This division not only fortifies the system against the vulnerability of a single point of failure but also significantly bolsters its fault tolerance. The system's resilience is such that an issue with a single device is isolated and does not compromise the overall stability of operations.

In the system, the worker is responsible for the training process of the model and uses the paillier homomorphic encryption algorithm to encrypt the model parameters, ensuring the security of data during transmission and preventing potential malicious attackers from stealing data. The validator is responsible for verifying the encrypted model parameters, effectively preventing malicious nodes from submitting harmful model updates. The miner is responsible for the maintenance of the blockchain and the addition of blocks, a process that provides the system with a stable and tamper-proof source of data.

By meticulously assigning roles, the system adeptly sidesteps the perils of single points of failure, substantially fortifying its resilience. This strategic approach guarantees that the integrity and continuity of the system's operation remain uncompromised, even in the face of individual device malfunctions. Furthermore, this role delineation not only elevates the system's operational efficiency and ease of management but also imbues it with heightened security and transparency, propelling the federated learning process towards greater efficacy and reliability.

Moving forward, we will provide a detailed introduction to the specific tasks and responsibilities of each role.

- worker: In the TPE-BFL system, the local model's training is the responsibility of the worker node. After the completion of training, the updated model parameters are securely encrypted using the Paillier encryption algorithm. These encrypted parameters are subsequently transmitted to the validator node, guaranteeing protection against potential theft by malicious attackers throughout the transmission process.
- validator: Upon receiving the encrypted model parameters, the validator utilizes the private key to verify the validity of these parameters.
- miner: The primary responsibility of the miner is to construct and validate blocks. Once the validator verifies the model, the validation results and voting outcomes are transmitted to the miner. The miner then validates them, packages the verification results into blocks, and subsequently adds these blocks to the blockchain.

In Fig. 1 we have detailed the flow of the entire TPE-BFL system, now we will cover steps ①-⑧ in Fig. 1 in detail. Before the whole workflow starts, we first carry out step ①, randomly divide all devices into three roles: worker, validator and miner. The devices of each role perform different tasks. Steps ②-③ show the work content of the worker node. In every communication round, the worker node modifies the local model. Prior to transmitting the local model parameters to the validator node, it is necessary for the worker node to encrypt them using the public key of the paillier homomorphic encryption algorithm. After completing this task, the worker will get a certain reward. Our

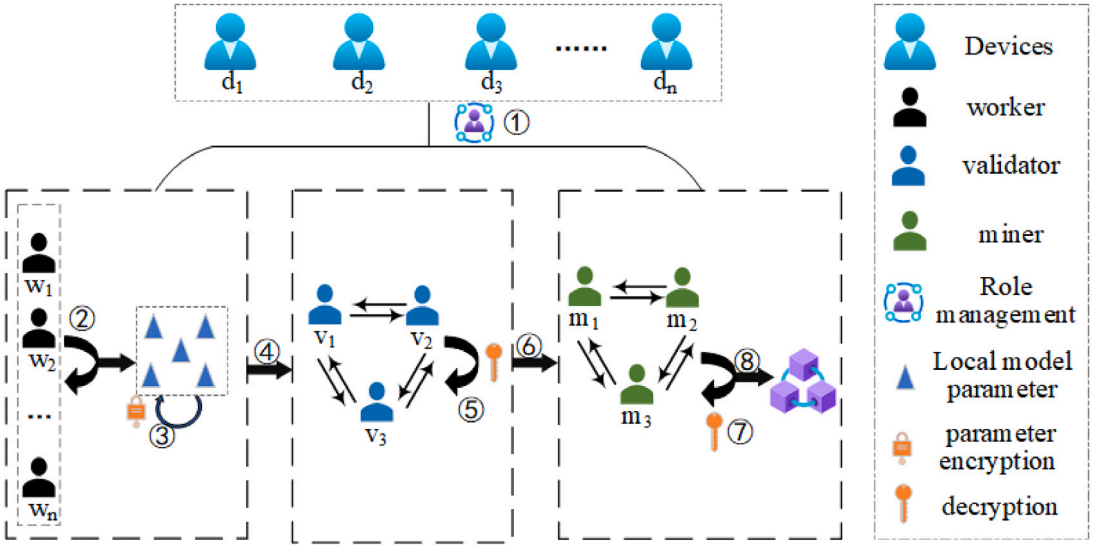


Fig. 1. TPE-BFL Framework.

Table 1
Mathematical notations in this work.

Symbol	Description
d	Device composition in TPE-BFL, $d \in D$;
w	Worker devices, $w \in W$;
v	Validator devices, $v \in V$;
m	Miner devices, $m \in M$;
R_i	The ith communication round;
L_{w_i}	The local model trained by worker;
θ_i	Model training parameter;
$\theta_i^{encrypt}$	The local model training parameter encrypted by paillier;
$\theta_i^{decrypt}$	The local model parameter decrypted by paillier;
G_i	The construction of the global model;
R_{w_i}	Rewards to worker in round R_i;
Tr_{w_i}	Worker-transaction containing L_{w_i}; $\theta_i^{encrypt}$; R_{w_i};
$R_{v_i}^{verify}$	Verification-reward by verified the worker-transaction in R_i;
$R_{v_i}^{validate}$	Voting-reward to validator in round R_i;
$v_i(\theta_i^{decrypt})$	Voting result;
Tr_{v_i}	Validator-transaction containing $R_{v_i}^{verify}$; $R_{v_i}^{validate}$;Tr_{w_i}; $v_i(\theta_i^{decrypt})$;
R_{m_i}	Miner-rewards
$block_i^m$	The miner m initiate the candidate block in round R_i;

proposed TPE-BFL uses a POS consensus mechanism to reward devices for completing tasks. After the reward is received, the worker encapsulates the encrypted local model and the obtained reward value into a transaction signed by the worker's public key.

In step ④, the worker sends the result of the transaction to its neighboring validator. Step ⑤ is the work content of the validator. After receiving the transaction result uploaded from the worker node, the validator broadcasts it to other validators. The validator employs the paillier private key for casting votes on the encrypted local model parameters after decrypting them. If the vote record of the Validator summary recorded in the legal block of Round R_i is positive, the worker is rewarded. After verifying the signature of the worker transaction, the validator will also get a verification reward, and extract $\theta_i^{encrypt}$ after verification for voting. The voting results are divided into positive and negative. A reward is given after the validator votes to verify the results. The merge is then encapsulated in the validator transaction.

In step ⑥, the encapsulated validator validation is transmitted to the miner. In step ⑦, after the miner obtains the validator transaction and the corresponding voting result from the validator, it uses the private key of the Paillier algorithm to decrypt the encapsulated validator transaction result. Ensure that only validated and published models are used for global model updates. In step ⑧, The miner creates a fresh block that includes all the legitimate updates to the encryption model, voting outcomes, and records of rewards. These newly generated blocks are candidate blocks. New blocks are mined using the POS consensus mechanism and added to the blockchain. In addition to the blockchain, the global model will be updated.

In our research, we have embraced the sophisticated design philosophy of blockchain as proposed by the VBFL framework [9]. The VBFL framework is an open and decentralized blockchain-based federated learning system that leverages miner nodes to generate new blocks. Building upon this foundation, we have integrated the paillier encryption algorithm to fortify the security of model parameters during transmission within the federated learning process. Specifically, following step ⑥ of the training process, miner nodes are tasked with creating a new block that encapsulates the encrypted model parameters. This cryptographic measure further ensures the security of training parameters throughout their transmission journey.

C. The TPE-BFL algorithm

In this section, we will provide a comprehensive explanation of the algorithmic description for our proposed approach. The central aspect of our method involves utilizing the paillier algorithm to encrypt the training parameters of the model, thereby guaranteeing their security during transmission.

Within the context of Algorithm 1, titled TPE-BFL, we integrate the Paillier cryptographic scheme into our federated learning architecture that operates on a blockchain foundation. The procedural steps 3 through 9 delineate the operational workflow for the worker nodes within the system. Upon the completion of local model training, these worker nodes proceed to encrypt the resultant training parameters utilizing the public key provided by the Paillier encryption methodology. Subsequently, the workers engage in the computation of the encrypted parameters $\theta_i^{encrypt}$ as prescribed by Formula (5). Following the successful execution of this process, the workers are conferred a reward R_{w_i} as an incentive for their participation and contribution to the federated learning process.

$$\begin{aligned} R_{w_i} &= Ne_{w_i} * |train_w| * r, if\, N_{v^+}(\theta_i^{decrypt}) \geq N_{v^-}(\theta_i^{decrypt}) \\ R_{w_i} &= 0, if\, N_{v^+}(\theta_i^{decrypt}) < N_{v^-}(\theta_i^{decrypt}) \end{aligned} \tag{8}$$

where Ne_{w_i} represents the number of local training epochs in R_i, $N_{v^+}(\theta_i^{decrypt})$ indicates the number of positive votes and $N_{v^-}(\theta_i^{decrypt})$ is the number of negative votes.

Lines 10 to 22 of the algorithm delineate the specific responsibilities of the validator within the system: Upon receiving a transaction Tr_{w_i} transmitted by worker, the validator verifies the transaction details and is accordingly rewarded for this verification task $R_{v_i}^{verify}$. Once

Algorithm 1: TPE-BFL

Input: Total number of devices N, Paillier Public-Private Key Pair: (PubKey, PrivKey)

1 **for** *each device i in N* **do**
2 randomly assigned role is $worker, validator, miner$
3 **for** *each CommunicationRound* R_i, $i \in (1, N)$ **do**
4 **for** *each Worker* **do**
5 Train Model with local data → Local Params θ_i;
6 PaillierEncrypt(PubKey, θ_i) →$(\theta_i^{encrypt})$;
7 $CalculateReward \rightarrow Reward(R_{w_i})$;
8 CreateTransaction Tr_{w_i}=$(\theta_i^{encrypt}, L_{w_i}, R_{w_i})$;
9 Send Tr_{w_i} to Validator;
10 **for** *each Validator* **do**
11 **while** *receiving Transaction Result* Tr_{w_i} *from Worker* **do**
12 Broadcast(Tr_{w_i});
13 **if** *VerifySignature* Tr_{w_i} *is Successful GrantReward* $R_{v_i}^{verify}$ **then**
14 Extract $\theta_i^{encrypt}$;
15 **else**
16 PaillierDecrypt(PrivKey, $\theta_i^{encrypt}$) → $\theta_i^{decrypt}$;
17 Verify and Vote$\theta_i^{decrypt}$ → VoteResult $v_i(\theta_i^{decrypt})$;
18 **if** *VoteResult is Positive* **then**
19 GrantReward $R_{v_i}^{validate}$;
20 **else**
21 Calculate Verification Reward → R_{v_i};
22 CreateTransaction Tr_{v_i}=$(R_{v_i}^{verify}, R_{v_i}^{validate}, v_i(\theta_i^{decrypt}), Tr_{w_i})$;
23 **for** *each Miner* **do**
24 **while** *receiving* Tr_{v_i} *from Validator* **do**
25 **if** *Validation Result is Valid* **then**
26 PaillierDecrypt (PrivKey, $Tr_{v_i}(\theta_i^{decrypt})$)
27 GrantReward R_{m_i};
28 CreateNewBlock;
29 Mine New Block using POS Consensus Mechanism;
30 Add New Block to Blockchain;
31 UpdateModelGlobalModel, NewBlock's Model Updates;
32 **else**

verification is successfully completed, the validator extracts the training parameters $\theta_i^{encrypt}$ from the transaction, which are in an encrypted state at this point. Using $\Pr ivKey$ and applying Formula (6), the validator decrypts these training parameters to obtain $\theta_i^{decrypt}$. Subsequently, the validator casts a vote on the decrypted training parameters $\theta_i^{decrypt}$ in accordance with the Proof-of-Stake (PoS) consensus mechanism and reaps additional rewards $R_{v_i}^{validate}$ for participating in the voting process.

$$R_{v_i}^{verify} = \{Tr_{w_i}\} * r \tag{9}$$

$$R_{v_i}^{validate} = \{v_i(\theta_i^{decrypt})\} * r \tag{10}$$

$$R_{v_i} = R_{v_i}^{verify} + R_{v_i}^{validate} \tag{11}$$

Lines 23–32 of the algorithm describe the specific work of miner, miner first checks the received verifier transaction Tr_{v_i}, uses the private key to decrypt the encryption parameters in the transaction, and also gets a return R_{m_i} after the verification is finished. The verified results are then used to update the global model, generating a block containing all valid encrypted model updates and voting records as well as reward records, and finally using the POS consensus mechanism to mine new blocks and add them to the blockchain.

$$R_{m_i} = |\{Tr_{v_i}(\theta_i^{decrypt})\}| * r \tag{12}$$

4. Proof of security

In the proposed federated learning system based on blockchain, the paillier algorithm is utilized by each worker to encrypt the local model parameters after training. These encrypted parameters are subsequently transmitted to validators for verification. Similarly, the encapsulated information received by miner from validators is also the result after encryption. In this process, the worker encrypts the training parameters using the public key, and the validator and miner use the private key for partial decryption operations. As the paillier encryption system ensures semantic security, the proposed blockchain-based federated learning system incorporates a training parameter encryption mechanism that effectively safeguards the privacy of transmitted training parameters.

Theorem. *According to the decision compound residual assumption (DCRA), the privacy of local model training parameters can be safeguarded by the proposed blockchain-based federated learning system's encryption mechanism.*

Proof. The proposed training parameter encryption mechanism's security relies on the utilization of the paillier encryption system. If the semantic security of this encryption system is ensured, then the blockchain-based federated learning system introduced in this paper will also possess semantic security.

It is assumed that there is an adversary A that can compromise the semantic security of the Paillier encryption system, and thus have access to private data.

Step1. Initialization: system generates public and private key pairs $(PubKey, PrivKey)$, and the $PubKey$ is distributed to all workers.

Step2. Local model training: Each worker uses local data to train local models L_{w_i} and generates training parameters θ_i.

Step3. Encryption transmission: The worker utilizes a $PubKey$ to encrypt the training parameters generated, resulting in $\theta_i^{encrypt}$, which is then transmitted to the validator.

$$\theta_i^{encrypt} = E(\theta_i, N, g) = g^{\theta_i} \cdot r^n \bmod N^2 \tag{13}$$

Step4. Validation selection: The validator randomly selects two sets of encrypted training parameters, denoted as $\theta_1^{encrypt}$ and $\theta_2^{encrypt}$, which may originate from different workers.

Step5. Decryption verification: The validator employs $\Pr ivKey$ to decrypt the received encrypted training parameters, thereby obtaining the plaintext $\theta_i^{decrypt}$, and subsequently verifies its legitimacy.

$$\theta_i^{decrypt} = D(\theta_i^{encrypt}, \lambda, N) = L((\theta_i^{encrypt})^{\lambda} \bmod N^2) \cdot \mu \bmod N \tag{14}$$

Step6. Guessing challenge: Adversary A attempts to distinguish whether $\theta_1^{encrypt}$ and $\theta_2^{encrypt}$ are encrypted training parameters derived from θ_1 or θ_2.

Step7. If adversary A is unable to distinguish between $\theta_1^{encrypt}$ and $\theta_2^{encrypt}$ with non-negligible advantage ε, then the system maintains privacy during the transmission process.

Construct a new adversary A′, who can utilize adversary A to undermine the semantic security of the Paillier encryption scheme. The attack strategy of adversary A′ involves:

a. A′ accepts the challenge of A and can obtain the paillier $PubKey$.

b. Adversary A′ generates two distinct legitimate model training parameters θ_1 and θ_2. Then encrypt the two training parameters to get $\theta_1^{encrypt}$, $\theta_2^{encrypt}$. And the two encryption parameters are sent to A.

$$\theta_1^{encrypt} = E(\theta_1, N, g) = g^{\theta_1} \cdot r^n \bmod N^2 \tag{15}$$

Table 2
Introduction of datasets.

Dataset	Train	Test	Size	Class
MNIST	60 000	10 000	28*28	10
Fashion-MNIST	60 000	10 000	28*28	10

$$\theta_2^{encrypt} = E(\theta_2, N, g) = g^{\theta_2} \cdot r^n \bmod N^2 \tag{16}$$

Step8. Adversary A attempts to distinguish between $\theta_1^{encrypt}$ and $\theta_2^{encrypt}$, and subsequently outputs a guess, denoted as b, indicating whether $\theta_1^{encrypt}$ and $\theta_2^{encrypt}$ are encrypted from θ_1 or θ_2.

Step9. If adversary A can distinguish between $\theta_1^{encrypt}$ and $\theta_2^{encrypt}$ with an advantage of ε, then adversary A′ can also make a guess with the same advantage ε regarding the origin of b.

Step10. According to the Decisional Composite Residuosity Assumption (DCRA): Even when provided with two randomly chosen elements g^a and g^b, an adversary cannot distinguish between g^{a+b} and g^r in polynomial time, where a+$b \equiv r \bmod n$. There exists no adversary, denoted as A′, capable of differentiating Paillier encryptions with non-negligible advantage.

Consequently, adversary A is also unable to distinguish between encrypted legitimate model parameters with non-negligible advantage ε, thereby substantiating the privacy of the system.

5. Experiments

In this section, we employ two publicly available datasets, namely MNIST and Fashion-MNIST, to empirically verify the effectiveness of the Training Parameter Encryption scheme in our proposed Blockchain Federated Learning system (TPE-BFL). The security of parameter transmission is guaranteed by encrypting training parameters in TPE-BFL. In this section's experiment, the model's performance will be assessed by progressively augmenting the quantity of malevolent nodes. The subsequent details pertaining to the procedure will be expounded upon, encompassing both experimental configuration and analysis.

A. Experimental Settings

Setup: Our experiments are conducted on a machine equipped with an Intel(R) Xeon(R) Gold 6326 CPU running at a speed of 2.90 GHz, along with an NVIDIA GTX 3090 Ti. The software environment consists of Python version 3.7.0 and Pytorch version 1.13.1. To validate our experimental results, we utilize two widely used datasets, namely MNIST and Fashion-MNIST.

MNIST is a dataset first proposed by Deng et al. [33] in 1998. This dataset comprises 10 distinct categories of handwritten numerals, encompassing a grand total of 70,000 instances. Among these, 60,000 samples are allocated for training purposes while the remaining 10,000 serve as testing data. The MNIST dataset, as a benchmark dataset has been widely used in experimental validation of machine learning. The Fashion-MNIST dataset is a dataset proposed by Han Xiao et al. [34], it used in this study is similar to the MNIST dataset in terms of image size, data format, and training and testing set structure. It comprises 70,000 grayscale images of fashion products from 10 different categories, each with a dimension of 28 * 28 pixels, which comes from different gender groups: men, women, children, and neutral. Each category has 7000 images. The Fashion-MNIST is divided into two categories for machine learning benchmark testing: training set and testing set, with 60 000 and 10 000 images respectively. Table 2 provides an introduction of datasets.

Implementation: We have designated a total of 100 communication rounds for our model validation process, utilizing a collective of 20 devices. Each local training session will proceed with a batch size of 10, employing a learning rate set to 0.01. And local train epoch set as 5. To further refine our approach, we have apportioned the roles within our system as follows: 12 workers, 5 validators, and 3 miners for the first configuration; and 8 workers, 9 validators, and 3 miners for the second. By varying the allocation of these roles, we intend to assess the impact on the model's performance. Our validation will involve training a Convolutional Neural Network (CNN) across two distinct datasets: MNIST and Fashion-MNIST. In addition, to better evaluate our approach, we dynamically adjust the number of malicious nodes to 0, 3 and 5.

Baselines: We conduct an equitable comparison of the model accuracy for TPE-BFL across varying levels of malicious node infiltration—specifically, when the count of such nodes is 0, 3 and 5. Additionally, our analysis extends to the examination of model performance under different distributions of participant roles. In this context, we will compare TPE-BFL with two seminal models: Vanilla FL and VBFL.

The Vanilla FL method [19], a cornerstone in the domain of federated learning, is predicated on the principle of retaining training data at the client level. It operates by amalgamating locally computed updates to foster the learning of a shared model.

VBFL [9] emerges as an augmented framework within federated learning. It integrates blockchain technology to remediate the vulnerabilities associated with centralized architectures, which are prevalent in traditional federated learning scenarios. These vulnerabilities include the susceptibility to single points of failure and the inability to authenticate the legitimacy of local models, thereby preventing the potential for malevolent devices to sabotage the model training process through nefarious activities.

B. Experimental Analysis

Our experimental evaluation will be conducted in two dimensions: initially, we will analyze the robustness of the TPE-BFL model; subsequently, we will conduct a comparative evaluation of the model's effectiveness. The robustness assessment of TPE-BFL will be executed across two distinct datasets, MNIST and Fashion-MNIST, with varying quantities of malicious nodes denoted as 0, 3, and 5. Additionally, the model will be evaluated under conditions where malicious nodes are present at a ratio of 15% of the overall participant count, with different allocations of participant roles. The comparative analysis will predominantly employ the MNIST dataset to contrast the model's efficacy from the vantage point of diverse configurations of malicious node quantities.

Model robustness analysis.

We mainly analyze the robustness of TPE-BFL model from the following two aspects. Firstly, we set up different number of malicious nodes from the point of view of the number of malicious nodes 0, 3, 5, namely 0%, 15% and 20% of the total number of all devices; As the quantity of malevolent nodes grows, we examine the effectiveness of our suggested approach TPE-BFL in aggregating global models. We use MNIST and Fashion-MNIST for experimental verification. The experiment conduct three sets of 100 communication rounds for experimental validation and the precision attained by the global model, which was generated collectively by 20 devices after every communication round, was recorded. In these 100 communication rounds, we divide these 20 devices into three roles: worker, validator and miner, with the number of roles being 12, 5 and 3.

Figs. 2 and 3 show the accuracy rate of TPE-BFL in global model training, verified in data sets: MNIST and FashionMNIST respectively, and adjusted the number of malicious nodes to 0, 3 and 5 when the number of worker, validator and miner nodes is set to 12, 5 and 3. The performance of TPE-BFL is assessed by gradually increasing the quantity of malicious nodes. Table 3 demonstrates that even with the introduction of malicious devices, our proposed TPE-BFL maintains a high level of accuracy during model training.

Second, when validating TPE-BFL, we divided 20 devices into three roles for validation. In the above experiment, we divided 20 devices into 12 workers, 5 validators, and 3 miners. However, dividing the number of different characters can also have a certain impact on the experiment. Next, we refer to the author's idea of role number

Table 3
TPE-BFL verified in MNIST and Fashion-MNIST(nm=0, 3, 5; worker, validator, miner=12, 5, 3).

Dataset	nm = 0	nm = 3	nm = 5
MNIST	0.91	0.92	0.86
Fashion-MNIST	0.88	0.75	0.72

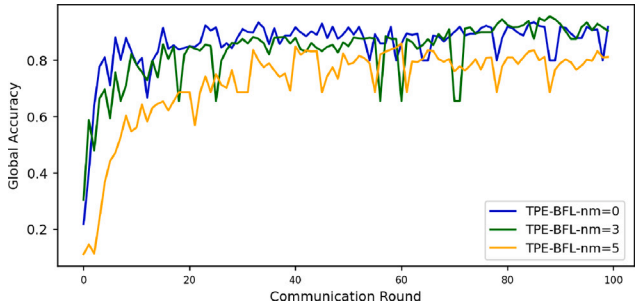


Fig. 2. Comparison in the MNIST dataset with the parameter setting as worker, validator, miner = 12, 5, 3; nm = 0, 3, 5.

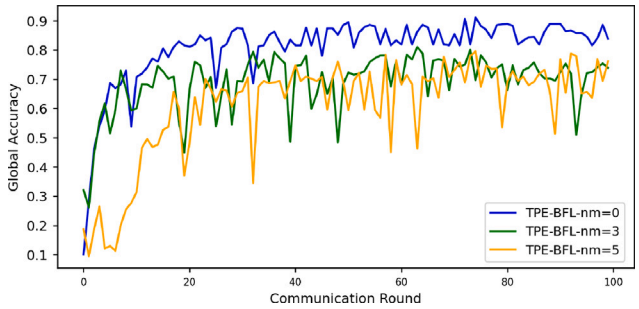


Fig. 3. Comparison in the FashionMNIST dataset with the parameter setting as worker, validator, miner = 12, 5, 3; nm = 0, 3, 5.

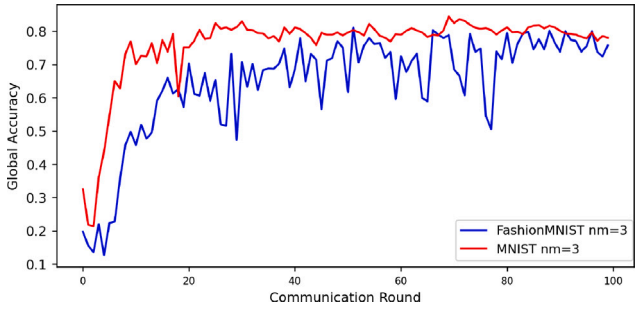


Fig. 4. Comparison with the same malicious devices (nm = 3) in MNIST and FashionMNIST dataset. The parameter setting is worker, validator, miner = 8, 9, 3.

Table 4
TPE-BFL verified in MNIST and Fashion-MNIST(nm=3; worker, validator, miner=8, 9, 3).

Dataset	nm = 3
MNIST	0.81
Fashion-MNIST	0.79

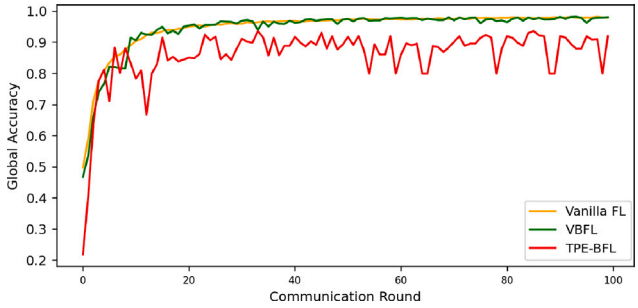


Fig. 5. Comparison with Vanilla FL, VBFL and TPE-BFL. The parameter nm = 0, worker, validator, miner = 12, 5, 3.

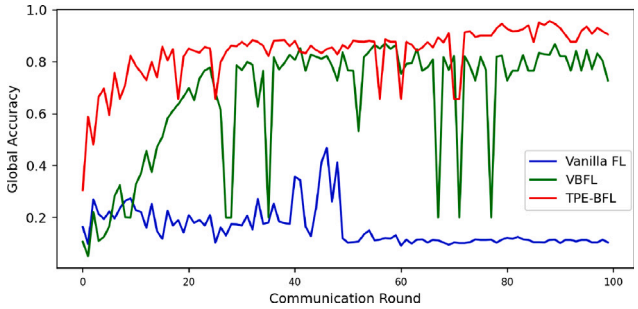


Fig. 6. Comparison with Vanilla FL, VBFL and TPE-BFL. The parameter nm = 3, worker, validator, miner = 12, 5, 3.

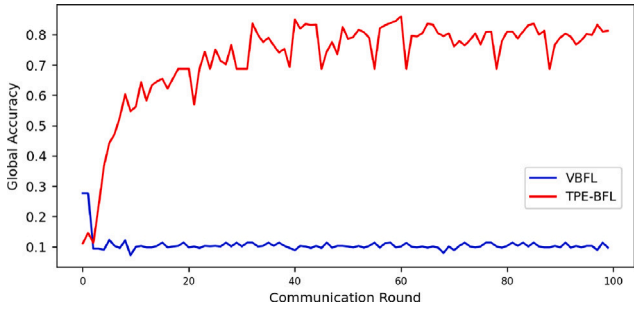


Fig. 7. Comparison with VBFL and TPE-BFL. The parameter nm = 5, worker, validator, miner = 12, 5, 3.

division in [35], adjust the three roles number to 8,9,3. Furthermore, we investigated how altering role numbers affects overall model accuracy. Experimental trials were performed on MNIST and FashionMNIST datasets (Fig. 4), while maintaining a constant total count of learning devices (20) and malicious nodes (3). The remaining parameter configurations remained unaltered to assess their influence on model precision. As shown in Table 4, our method can still maintain high accuracy on both data sets. This result shows that TPE-BFL is robust.

Model comparison analysis.

In this part, we compare TPE-BFL with Vanillia FL and VBFL models. The comparison experiment mainly uses MNIST data set to conduct comparative analysis from the perspective of setting different numbers of malicious nodes.

As depicted in Fig. 5, it is evident that the accuracy of the global model remains unaffected by the absence of any malicious devices when employing the proposed approach. That is to say, the incorporation of the paillier algorithm will have no impact on the formation of the overall model. When the number of malicious devices is 15% of the total number, there are 3 malicious devices. According to the data presented in Fig. 6, TPE-BFL exhibits superior accuracy compared to VBFL and Vaillia FL, with a model accuracy of 0.926. Fig. 7 illustrates that when the proportion of malicious devices reaches 20% of the overall count, specifically accounting for 5 devices, it becomes evident that during this period, the VBFL model exhibits significantly diminished accuracy in constructing the global model, only 0.103, while the TPE-BFL proposed by us still maintains a stable state of 0.86. The experimental results show that when there are malicious

devices, the accuracy of Vaillia FL and VBFL in constructing global models is seriously affected, and TPE-BFL with the addition of paillier homomorphic encryption algorithm has better performance.TPE-BFL uses paillier algorithm to encrypt the transmitted training parameters to ensure the security of the training parameters. The experimental findings above demonstrate that the presence of malicious nodes does not impact the model's performance, thanks to the encryption applied to training parameters. Consequently, a high level of accuracy can be sustained.

6. Conclusion

In this article, we introduce a novel approach called Training Parameter Encryption scheme for Blockchain based Federated Learning System (TPE-BFL). Unlike conventional blockchain federated learning frameworks, our method employs the paillier encryption algorithm to secure the transmission of local model parameters during the training process across worker, validator, and miner nodes. By utilizing encrypted model parameters, we successfully train a global model while ensuring data security. Experimental evaluation conducted on MNIST and Fashion-MNIST datasets demonstrates the effectiveness and accuracy of our proposed TPE-BFL.

In the future of our work, communication efficiency and computing overhead need to be emphasized. We plan to adjust the POS consensus mechanism and choose a consensus algorithm with less computational overhead to improve the efficiency of the model.

CRediT authorship contribution statement

Fanfan Shen: Methodology. **Qiwei Liang:** Writing – original draft. **Lijie Hui:** Writing – review & editing. **Bofan Yang:** Validation. **Chao Xu:** Formal analysis. **Jun Feng:** Conceptualization. **Yanxiang He:** Project administration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

Our work is supported by the Basic Science (Natural Science) Research Project of Colleges and Universities in Jiangsu Province, China (22KJA520004, 20KJA520002), the National Natural Science Foundation of China (61902189, 71972102, 61972293, 62372195), the Excellent Science and Technology Innovation Team Project in Jiangsu Province's Universities (2021).

References

[1] Olivia Choudhury, Aris Gkoulalas-Divanis, Theodoros Salonidis, Issa Sylla, Yoonyoung Park, Grace Hsu, Amar Das, Differential privacy-enabled federated learning for sensitive health data, 2019, arXiv preprint arXiv:1910.02578.

[2] Chen Zhang, Yu Xie, Hang Bai, Bin Yu, Weihong Li, Yuan Gao, A survey on federated learning, Knowl.-Based Syst. 216 (2021) 106775.

[3] Qinbin Li, Zeyi Wen, Zhaomin Wu, Sixu Hu, Naibo Wang, Yuan Li, Xu Liu, Bingsheng He, A survey on federated learning systems: Vision, hype and reality for data privacy and protection, IEEE Trans. Knowl. Data Eng. (2021).

[4] Qiang Yang, Yang Liu, Tianjian Chen, Yongxin Tong, Federated machine learning: Concept and applications, ACM Trans. Intell. Syst. Technol. 10 (2) (2019) 1–19.

[5] Weishan Zhang, Qinghua Lu, Qiuyu Yu, Zhaotong Li, Yue Liu, Sin Kit Lo, Shiping Chen, Xiwei Xu, Liming Zhu, Blockchain-based federated learning for device failure detection in industrial iot, IEEE Internet Things J. 8 (7) (2020) 5926–5937.

[6] Theodora S. Brisimi, Ruidi Chen, Theofanie Mela, Alex Olshevsky, Ioannis Ch. Paschalidis, Wei Shi, Federated learning of predictive models from federated electronic health records, Int. J. Med. Inform. 112 (2018) 59–67.

[7] Chuan Ma, Jun Li, Long Shi, Ming Ding, Taotao Wang, Zhu Han, H. Vincent Poor, When federated learning meets blockchain: A new distributed learning paradigm, IEEE Comput. Intell. Mag. 17 (3) (2022) 26–33.

[8] Hyesung Kim, Jihong Park, Mehdi Bennis, Seong-Lyun Kim, Blockchained on-device federated learning, IEEE Commun. Lett. 24 (6) (2019) 1279–1283.

[9] Hang Chen, Syed Ali Asif, Jihong Park, Chien-Chung Shen, Mehdi Bennis, Robust blockchained federated learning with model validation and proof-of-stake inspired consensus, 2021, arXiv preprint arXiv:2101.03300.

[10] Wael Issa, Nour Moustafa, Benjamin Turnbull, Nasrin Sohrabi, Zahir Tari, Blockchain-based federated learning for securing internet of things: A comprehensive survey, ACM Comput. Surv. 55 (9) (2023) 1–43.

[11] Sana Awan, Fengjun Li, Bo Luo, Mei 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, CCS '19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 2561–2563.

[12] Mochan Fan, Kailai Ji, Zhaofeng Zhang, Hongfang Yu, Gang Sun, Lightweight privacy and security computing for blockchained federated learning in iot, IEEE Internet Things J. (2023).

[13] Xiaohui Yang, Chongbo Xing, et al., Federated medical learning framework based on blockchain and homomorphic encryption, Wirel. Commun. Mob. Comput. (2024) 2024.

[14] Xianglong Zhang, Anmin Fu, Huaqun Wang, Chunyi Zhou, Zhenzhu Chen, A privacy-preserving and verifiable federated learning scheme, in: ICC 2020-2020 IEEE International Conference on Communications, ICC, IEEE, 2020, pp. 1–6.

[15] Truc Nguyen, My T. Thai, Preserving privacy and security in federated learning, IEEE/ACM Trans. Netw. (2023).

[16] Muhammad Shayan, Clement Fung, Chris J.M. Yoon, Ivan Beschastnikh, Biscotti: A blockchain system for private and secure federated learning, IEEE Trans. Parallel Distrib. Syst. 32 (7) (2021) 1513–1525.

[17] Bin Jia, Xiaosong Zhang, Jiewen Liu, Yang Zhang, Ke Huang, Yongquan Liang, Blockchain-enabled federated learning data protection aggregation scheme with differential privacy and homomorphic encryption in iiot, IEEE Trans. Ind. Inform. 18 (6) (2022) 4049–4058.

[18] Yuanhang Qi, M. Shamim Hossain, Jiangtian Nie, Xuandi Li, Privacy-preserving blockchain-based federated learning for traffic flow prediction, Future Gener. Comput. Syst. 117 (2021) 328–337.

[19] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, Blaise Aguera y Arcas, Communication-efficient learning of deep networks from decentralized data, in: Artificial Intelligence and Statistics, PMLR, 2017, pp. 1273–1282.

[20] Meng Hao, Hongwei Li, Xizhao Luo, Guowen Xu, Haomiao Yang, Sen Liu, Efficient and privacy-enhanced federated learning for industrial artificial intelligence, IEEE Trans. Ind. Inform. 16 (10) (2020) 6532–6542.

[21] Jakub Konečnỳ, H.Brendan McMahan, Felix.X Yu, Peter Richtárik, Ananda Theertha Suresh, Dave Bacon, Federated learning: Strategies for improving communication efficiency, 2016, arXiv preprint arXiv:1610.05492.

[22] Caner Korkmaz, Halil Eralp Kocas, Ahmet Uysal, Ahmed Masry, Oznur Ozkasap, Baris Akgun, Chain fl: Decentralized federated machine learning via blockchain, in: 2020 Second International Conference on Blockchain Computing and Applications, BCCA, IEEE, 2020, pp. 140–146.

[23] Irshad Ullah, Xiaoheng Deng, Xinjun Pei, Ping Jiang, Husnain Mushtaq, A verifiable and privacy-preserving blockchain-based federated learning approach, Peer-to-Peer Netw. Appl. 16 (5) (2023) 2256–2270.

[24] Saja J. Mohammed, Dujan B. Taha, Paillier cryptosystem enhancement for homomorphic encryption technique, Multimedia Tools Appl. 83 (8) (2024) 22567–22579.

[25] Saja J. Mohammed, Dujan B. Taha, Performance evaluation of rsa, elgamal, and paillier partial homomorphic encryption algorithms, in: 2022 International Conference on Computer Science and Software Engineering, CSASE, IEEE, 2022, pp. 89–94.

[26] Chiara Marcolla, Victor Sucasas, Marc Manzano, Riccardo Bassoli, Frank HP Fitzek, Najwa Aaraj, Survey on fully homomorphic encryption, theory, and applications, Proc. IEEE 110 (10) (2022) 1572–1609.

[27] Pascal Paillier, Public-key cryptosystems based on composite degree residuosity classes, in: International Conference on the Theory and Applications of Cryptographic Techniques, Springer, 1999, pp. 223–238.

[28] Haokun Fang, Quan Qian, Privacy preserving machine learning with homomorphic encryption and federated learning, Future Internet 13 (4) (2021) 94.

[29] Xianglong Zhang, Anmin Fu, Huaqun Wang, Chunyi Zhou, Zhenzhu Chen, A privacy-preserving and verifiable federated learning scheme, in: ICC 2020-2020 IEEE International Conference on Communications, ICC, 2020, pp. 1–6.

[30] Chengliang Zhang, Suyi Li, Junzhe Xia, Wei Wang, Feng Yan, Yang Liu, BatchCrypt: Efficient homomorphic encryption for Cross-Silo federated learning, in: 2020 USENIX Annual Technical Conference (USENIX ATC 20), USENIX Association, 2020, pp. 493–506.

[31] Xu Ma, Yuqing Zhou, Laihua Wang, Meixia Miao, Privacy-preserving byzantine-robust federated learning, Comput. Stand. Interfaces 80 (2022) 103561.
[32] Wenqiang Yang, Bin Liu, Changlei Lu, Nenghai Yu, Privacy preserving on updated parameters in federated learning, in: Proceedings of the ACM Turing Celebration Conference-China, 2020, pp. 27–31.
[33] Li Deng, The mnist database of handwritten digit images for machine learning research [best of the web], IEEE Signal Process. Mag. 29 (6) (2012) 141–142.
[34] Han Xiao, Kashif Rasul, Roland Vollgraf, Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, 2017, arXiv preprint arXiv:1708.07747.
[35] Feng Yu, Hui Lin, Xiaoding Wang, Abdussalam Yassine, M. Shamim Hossain, Blockchain-empowered secure federated learning system: Architecture and applications, Comput. Commun. 196 (2022) 55–65.

Fanfan Shen received the Ph.D. degree from Wuhan University, Hubei, China. He is now a fulltime associate professor at the Nanjing Audit University, Jiangsu, China. His main research interests include computer system and artificial intelligence (ffshen@whu.edu.cn).

Qiwei Liang graduated from Jinzhong University in 2021 with a Bachelor's degree in Data Science and Big Data. She is currently pursuing a master's degree at Nanjing Audit University. Her main research interests include federated learning and privacy protection.

Lijie Hui graduated from Henan University of Urban Construction in 2022 with a Bachelor's degree in Computer Science and Technology. She is currently pursuing a master's degree at Nanjing Audit University. Her main research interests include federated learning and privacy protection.

Bofan Yang graduated from Henan University of Technology with a bachelor's degree in IOT engineering in 2021. He is currently pursuing amaster's degree at Nanjing Audit University. His main research interests include UAV path planning and MEC networks.

Chao Xu received the BS and Ph.D. degrees from the Computer School at the Wuhan University, Hubei, China. He is now a professor at Nanjing Auditing University, China. His main research interests include trust computing and artificial intelligence.

Jun Feng (Member, IEEE) received the Ph.D. degree from the School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China. He is an associate professor with the School of Cyber Science and Engineering, Huazhong University of Science and Technology. He is particularly interested in privacy-preserving machine learning, cyber–physical-social systems, differential privacy, deep learning, blockchain, and Big Data.

Yanxiang He received the Ph.D. degree from Wuhan University, Hubei, China. He is a Professor in the Computer School at the Wuhan University. His research interests include trusted software, distributed parallel processing and high performance computing.