Privacy Preservation in User Behavior Analysis for Mobile Edge Computing

Song Deng, *Member, IEEE* , Jie Zhang, Longxiang Zhang

***Abstract*—In mobile edge computing, user behavior analysis can provide a more clear understanding of edge user behavior, which is crucial for providing more accurate recommendation services to mobile users. However, privacy leakage is inevitable in the process of edge user behavior data collection, storage and analysis. How to protect edge user privacy is very urgent for the development of mobile edge computing. Existing schemes dealing with these problems still suffer from 1) high computational overhead, 2) low model accuracy and poor convergence under privacy-preserving, and 3) poor model robustness when edge nodes fail. Therefore, we propose a novel privacy-preserving model in edge user behavioral analysis termed Safe-FL to protect the behavioral privacy of all edge participants by introducing secure data aggregation and encryption in the federated learning (FL) framework. To reduce the privacy leakage of data uploading in edge user behavior analysis, we first construct a federated learning-based edge user behavior analysis framework. Secondly, we utilize a hybrid encryption scheme to encrypt the local model hierarchically to protect the gradient privacy in edge user behavior analysis and reduce the encryption computation overhead. And finally, to improve the robustness of the model, we propose a decentralized secure gradient aggregation algorithm based on blockchain. Extensive experiments are conducted on two public datasets, whose results illustrate that our model has 1) strong privacy preservation and small computational overhead; 2) high accuracy and convergence speed; and 3) good robustness, compared to other models.**

***Index Terms*—Mobile edge computing, Privacy preservation, User behavior analysis, Federated learning, Blockchain**

I. Introduction

As more and more edge mobile devices, such as smartphones, PCs, wearables, and other sensing devices, are connected to 5G or other wireless networks, edge smart terminals have become an integral part of people's daily lives and work [1]. By analyzing the huge amount of user behavior data produced by these edge terminals, we can better understand and satisfy the personalized needs of users. And efficient and accurate edge user behavior analysis is essential to gain insights into the actual needs, behavioral patterns and preferences of edge users and provide fast service response [2]. However, for terminals at the edge of the network, centralized processing based on cloud computing can hardly process the data generated by these edge devices efficiently and cannot meet the real-time behavior analysis of users.

In the edge computing setting, various types of user behavior data are scattered and stored on edge terminals. If the user behavior data on the edge terminals are processed centrally, it is necessary to transmit these data from the edge to the cloud server, which will undoubtedly increase the security risk during data transmission [3][4][5]. Federated learning can provide security privacy guarantees for user behavioral data executed locally at the edge [6][7][8]. It is able to process user data and model training in edge devices directly at the edge terminal, thus avoiding the leakage of user privacy caused by data transmission [9]. However, the ubiquitous and open nature of edge terminals makes them more vulnerable to cyber attacks [10]. Hackers are able to steal the behavioral data of edge users through information exploitation attacks, data poisoning attacks, model poisoning attacks, and hitchhiking attacks, which in turn cause user behavioral data leakage [11][12]. This poses a huge risk of privacy leakage for user behavior analysis in edge computing.

Although analyzing and processing data on edge servers closer to the user can meet the user's high responsive demand for services, it still faces the following challenges: 1) Most of the existing edge user behavior analysis focuses on reducing the service queuing delay, transmission cost, and communication delay of edge nodes, and does not consider the privacy protection of edge user behavior [13], which makes edge user behavior analysis easy to leak user privacy; 2) The introduction of edge computing in federated learning can better realize distributed, efficient, and secure model training [14], however, the parameter gradient uploaded through the client in federated learning may leak sensitive information [15][16]. Overall, there are still practical challenges to be addressed regarding data privacy itself and network transmission efficiency.

To handle the above challenges, we design a privacy preservation model in edge user behavior analysis, called Safe-FL model. The model introduces federated learning and blockchain technology in the edge computing environment, where the local model is first trained in the edge server, and finally, a hybrid encryption algorithm based on homomorphic encryption and differential privacy protects the model parameters and ensures the security of the model gradient, thus completing the secure aggregation to generate the global model. Blockchain is a decentralized and traceable distributed ledger that records every data interaction, enabling the history of data usage to be traced and verified. This transparency helps establish trust and allows participants to better supervise the use of data. Furthermore, the decentralized nature of blockchain ensures that data is

Manuscript received XXXX XX, 2022; revised XXXX XX, 2022; accepted XXXX XX, 2022. This work was supported by the National Natural Science Foundation of China under Grant No. 51977113, 62293500, 62293501 and 62293505.

Song Deng is Institute of Advanced Technology, Nanjing University of Posts and Telecommunications, Nanjing 210003, China (e-mail : dengsong@njupt.edu.cn)

Jie Zhang and Longxiang Zhang are with the College of Automation, Nanjing University of Posts and Telecommunications, Nanjing, 210003, China (e-mail: 18256797670@163.com, zhanglongxianggn@163.com).

stored across multiple nodes rather than on a centralized server [17]. This distributed structure provides higher security and resilience against attacks, reducing the risk of data tampering or deletion [18]. Lastly, blockchain technology can provide proof of compliance and legitimacy while ensuring data privacy.

In summary, main contributions of this paper are summarized as follows:

1) We apply federated learning to privacy preservation model in edge user behavior analysis (Safe-FL). The model allows a distributed training process without centralized uploading of user data to a cloud server.

2) In Safe-FL, we use a hybrid encryption scheme with guaranteed data privacy to achieve hierarchical encryption of the local model. This can protect gradient privacy and reduces the computation and communication overhead of encryption process.

3) And, we build a blockchain-based decentralized secure gradient aggregation execution environment that guarantees the security and accuracy of the model while improving the robustness of the model, and makes the model unaffected by edge node exits.

4) Extensive experiments on two real datasets are conducted, and the results illustrate that the superior performance of the proposed model over local model and federated learning model in terms of computation and communication overhead, accuracy, loss, and robustness.

The remainder of this paper is organized as follows. Section II reviews the related work. The preliminary concept is introduced in Section III. Section IV focuses on details of Safe-FL. Section V conducts the extensive experiments on two datasets. Section VI concludes this paper.

II. RELATED WORK

In this section, we review the pros and cons of existing models in terms of user behavior analysis and privacy protection. Firstly, we analyze the various models on user behavior analysis and privacy protection. And finally, our remark and solution are given.

A. User behavior analysis

User behavior analysis helps to discover user behavioral habits and correctly analyze user interests, thereby providing data support for user personalized services. Most user behavior analysis methods are mainly implemented through traditional machine learning and deep learning algorithms.

Complex and diverse user behaviors are high invisibility and initiative. Chen et al [19] applied K-Means clustering algorithm to user behavior analysis, but the accuracy is not high. Dai et al [20] proposed a user behavior clustering method based on association rule mining, which improves the clustering quality but the algorithm does not have feature selection capability. Xue et al [21] used LSTM to automatically extract and select features of users' consumption behavior data, and then predicted users' purchasing behavior by SVM. To improve the generalization ability of the algorithm, Hu et al [22] constructed a hybrid prediction model of user purchase behavior based on soft voting method with logistic regression and support vector machine. Gan et al [23] proposed a new model called R-RNN, which applies the attention mechanism to help capture the representation of users' main interests and combined it with LSTM to explore the interest trends behind users' recent clicking behavior. In [24], He et al. proposed an algorithm for extracting feature representation vectors of users and goods based on LightGCN model on user-goods interaction graph. The feature representation vector is used to aggregate the nodes in the interaction graph, which improves the recommendation effect of the model. However, only considering a single user's behavior cannot explore the potential preference of the user, Yu et al [25] use graph convolutional neural network to learn the user's preference for different single behaviors, and get the final recommendation result by fusion prediction.

Compared with traditional machine learning models, deep learning models are more expressive of nonlinearity, can automatically learn and extract features from raw data, and have strong generalization capabilities. However, existing user behavior analysis based on both machine learning and deep learning do not consider the privacy protection of user behavior data, and thus user behavior is easily exposed when performing user behavior analysis.

B. Privacy protection

User behavior analysis based on centralized training has a serious risk of privacy leakage. Federated learning can prevent the privacy leakage of centralized training. Therefore, researchers apply federated learning to edge computing, which can effectively protect the data privacy security of edge users [26].

Taik et al [27] constructed a distributed learning scheme based on edge computing and federated learning, which provided a diverse amount of data for training deep learning models and protected user privacy. Liu et al [28] discarded clients that exceeded the time limit in the aggregation phase by limiting the gradient upload time. Compared with the traditional FL algorithm, this method effectively reduces the communication overhead of the in-vehicle network. To reduce the computational cost and communication delay of mobile edge devices, Ye et al [29] designed the EdgeFed architecture based on edge computing and federated learning. The architecture decentralized the process of updating local models to edge servers, which greatly improves the learning efficiency.

However, the above schemes do not impose effective protection on the gradient parameters during the model aggregation process, and there is still a risk of user privacy leakage. In view of the user's privacy and the limited resources of the edge cloud, Qian et al [30] proposed a privacy-aware service placement (PSP) strategy, which can protect the user's privacy while providing the user with better service quality.To construct a secure federated learning scheme, Lu et al [31] applied local differential privacy to federated learning as a way to protect the privacy of the local model during model gradient aggregation. Zhou et al [32] designed a privacy-preserving method to protect clients against differential privacy noise in the proposed PVFL framework, and achieved a smaller

noise size than the traditional differential privacy mechanism. Jia et al [33] proposed distributed Kmeans clustering based on differential privacy and homomorphic encryption, as well as distributed Random Forest with differential privacy and distributed AdaBoost method with homomorphic encryption, and combined with blockchain to have good performance in real IOT application scenarios. He et al [34] utilized the improved Paillier homomorphic encryption algorithm for encrypted transmission of local gradient parameters of edge terminal devices, which reduces the encryption overhead and improves the efficiency while protecting privacy. Alkhelaiwi et al [35] proposed a partial homomorphic encryption scheme that can handle confidential information without exposing the underlying data. However, the ciphertext computation overhead is high in practical applications.

C. Federated Learning and Blockchain

The integration of federated learning and blockchain is an emerging research direction that has attracted considerable attention and exploration. Currently, research on the combination of federated learning and blockchain encompasses the following research directions: privacy preservation, security and tamper-resistance, and collaboration and incentive mechanisms based on smart contracts.

One of the main advantages of federated learning is conducting model training without sharing raw data, thus protecting the privacy of participants. However, to further enhance data privacy, researchers have begun incorporating blockchain technology into federated learning. By leveraging the distributed and verifiable properties of blockchain, transparency and privacy protection in data interactions can be ensured, thereby improving the privacy-preserving capabilities of federated learning systems [36]. Furthermore, since federated learning involves collaboration among multiple participants, security and tamper-resistance become critical concerns. The immutability and decentralized nature of blockchain can help ensure the validation and auditing of training updates, thus ensuring the trustworthiness of model updates submitted by participants [37]. Using blockchain in federated learning can also provide anti-attack and decentralized solutions, enhancing the security of the system. The smart contract functionality of blockchain can be utilized to define and execute collaboration protocols and incentive mechanisms among federated learning participants [38]. Smart contracts can automate the enforcement of interaction rules among participants, ensuring fairness and proper allocation of rewards.

Traditional deep learning involves storing the dataset centrally on a server and conducting model training on that server. However, this centralized approach may raise concerns about privacy and data security, especially when sensitive data includes personal information. On the other hand, federated learning is a distributed learning method that allows for model training while preserving data privacy. In federated learning, data is stored locally on devices, and model training takes place on those devices. Only the updated model parameters are sent to the central server, thus protecting the privacy of individual data.

III. Preliminary Concept

To describe how to build a privacy preservation model in edge user behavior analysis, in this section, we introduce the preliminary concepts and processes of federated learning [39], homomorphic encryption [40], differential privacy [41], and blockchain [42], respectively, and the roles they play in the model. The core technical framework is shown in Fig.1.

In the aggregation process, our Safe-FL model uses a hybrid encryption technique of full homomorphic encryption and differential privacy, and achieves gradient encryption by means of hierarchical optimization and secure aggregation of encrypted data. Meanwhile, to enhance network security, we use a decentralized blockchain network in the Safe-FL model to ensure the reliability and security of the aggregation process by migrating the aggregation computation process in the Safe-FL model to smart contracts in blockchain. In addition, the Safe-FL model also enhances the robustness of the model based on the traceability of the blockchain to increase the tolerance of node failures. To this end, we briefly introduce the basic concepts of these techniques.

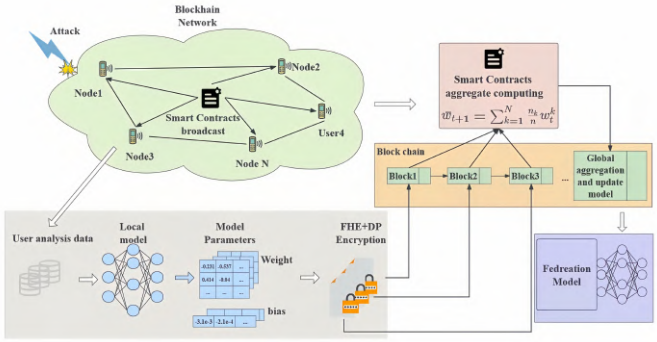

Fig. 1. Core technical framework.

A. Federated learning

Federated learning aims to build a learning model based on a distributed dataset, including model training and model inference. During model training, model-related information can be exchanged (or exchanged in an encrypted manner) between parties, but local private data cannot be exchanged. This exchange does not expose the privacy of each user's data. Let matrix D_i represent the data of the $i-th$ user. Each row of D_i represents a data sample, and each column represents a specific data feature. And, let the feature space be x , the data label be y, and denote the data sample ID space. Both feature space and sample ID space of data owned by different users may be different. According to the feature space and sample ID space distribution of different training data among different users, federated learning can be categorized into horizontal federated learning, vertical federated learning, and federated migration learning. Among them, horizontal federation learning, vertical federation learning and federation migration learning are denoted as follows.

$$\begin{aligned} x_i = x_j, y_i = y_j, \mathrm{I}_i \neq \mathrm{I}_j, \forall D_i, D_j, i \neq j \\ x_i \neq x_j, y_i \neq y_j, \mathrm{I}_i = \mathrm{I}_j, \forall D_i, D_j, i \neq j \\ x_i \neq x_j, y_i \neq y_j, \mathrm{I}_i \neq \mathrm{I}_j, \forall D_i, D_j, i \neq j \end{aligned} \tag{1}$$

B. Homomorphic encryption

Homomorphic encryption allows computations to be performed directly on the encrypted data without access to the key. We perform some basic mathematical operations on the ciphertext, and then decrypt the resulting ciphertext to produce the same result as the original plaintext with the same mathematical operations.

C. Differential privacy

In differential privacy, the function output result is insensitive to any particular record in the dataset. Hence it can be used to resist membership inference attacks. Differential privacy balances utility and privacy while introducing noise.

Definition 1. *Let two neighboring datasets D and $D^{'}$ differ by only one record, and M be a randomization algorithm obeying a certain distribution, for all $S \in Range(M)$, we have the following inequality holds.*

$$Pr[M(D) \in S] \leq Pr[M(D_{\prime}) \in S] \times \varrho^{\epsilon} + \delta \qquad (2)$$

Then the process is called (ε, δ) differential privacy, where ε denotes the privacy budget and δ denotes the failure probability.

D. Single point of failure

A single point of failure refers to a situation in a federated learning network where the normal operation of the entire system is affected due to the failure or malfunction of a specific node. This can be caused by hardware failures, power outages, network connectivity issues, and so on, resulting in the system's inability to function properly. If a critical node experiences a failure, it can disrupt the regular operation of the entire network. The occurrence of a single point of failure in federated learning can have significant consequences, such as model termination or unforeseen outcomes. Conversely, blockchain is designed to be a distributed system that disperses data and computational power across multiple nodes, enhancing system security and robustness.

IV. Privacy Preservation in Edge User Behavior Analysis

In this section, we first describe the detailed workflow and functionalities of our model. Then, we proceed to analyze and demonstrate the model's convergence.

A. System Processes

The entire detailed working processes of the Safe-FL model is shown in Fig.2.

a) System Initialization: When the blockchain based edge federated learning model starts working, the first step is to perform system initialization. The local node will initialize the local model and also generate the key pair (pk_i, sk_i) for encryption. These keys can be used both to encrypt the local model and to perform authentication to ensure data security. The local node will also set up the smart contract algorithm and consensus algorithm of the edge blockchain node to ensure

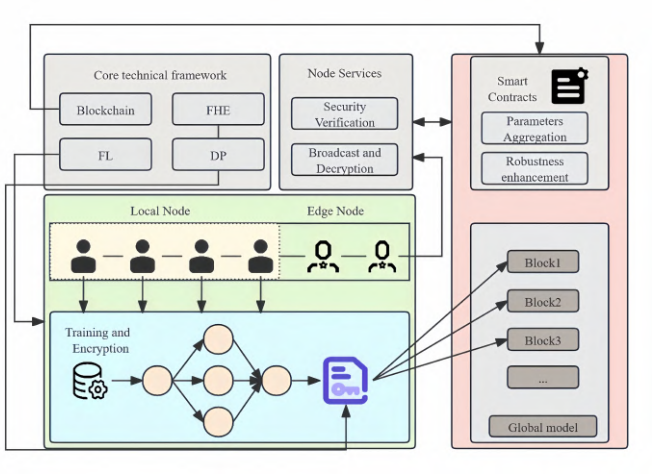

Fig. 2. Safe-FL workflow.

the compliance and stability of the system, thereby achieving a secure and trustworthy data exchange environment through smart contracts. System initialization is the foundation of the entire edge computing model. The local node first selects the appropriate encryption algorithm and key length and generates the key pair (pk_i, sk_i) including the public key pk_i and private key sk_i . The public key pk_i is used to encrypt the trained model parameters and the private key sk_i is used to decrypt the model parameters. Meanwhile, local nodes also need to set up the smart contract algorithm, which can define the way of collaboration between edge nodes and determine the strategy and rules for model training. Finally, the consensus algorithm is selected to ensure the consistency and reliability of the edge blockchain network.

b) Training and Encryption: Next, the local node start the training and encryption of the local model as shown in Algorithm 1. Local nodes $(P_1, P_2, ...P_N)$ completes the training using the local dataset $(D_1, D_2, ...D_N)$ and uses the optimization algorithm to update the parameters $(w_1, w_2, ...w_N)$ of the model. Iteratively carry out this process to continuously improve the performance of the model. After the completion of local training, the local node encrypts the model parameters to safeguard the model's confidential information while ensuring secure transmission of these parameters. In order to store the data on the edge blockchain node, the local node must access the interface of the edge blockchain. Before accessing the interface, the local node undergoes an authentication process to ensure that only nodes with the appropriate privileges can upload data. Authentication can be performed using generated key pairs for signature verification to ensure the identity and authority of the node. If $Identity(pk_i, sk_{sk_i}) = true$, the local node uploads the processed data to the edge blockchain node. The edge blockchain serves as a distributed network that guarantees data traceability and transparency. During the data uploading process, the local node's data is encapsulated into a transaction and subsequently uploaded to the blockchain. This approach allows all participants to access the transaction's details, including the identity information of the uploader, the content of the uploaded data, etc., thus enhancing the

transparency and credibility of the system.

$$[[w_n]] \leftarrow DPenc(w_{weight,r_{(n)}}) + FHEenc(pk_n, w_{bias}) \quad (3)$$

Here, $DPenc$ refers to Differential Privacy Encryption. $FHEenc$ refers to Fully Homomorphic Encryption. r represents the locally generated random noise at each node, while pk corresponds to the public key of the local node.

Algorithm 1: User local training and encryption

Input: $w^{(0)}$
Output: $[[w^{(n)}]](n \in SP)$
Data: ρ, S, M, λ, D_n, $SP((P_1, P_2, ...P_N) \in SP)$

Local training process ;
Initialization: $w_{1,1}^{(n)} = w^{(0)}$;
for $n \in SP$ **do**
  for $i = 1$**to**S **do**
    $Batches = \frac{D_n}{M}$;
    for $b = 1$**to**$Batches$ **do**
      $g_n = \nabla \frac{1}{C_n} \sum_{i \in SP} f_i(w)$;
      $w_{b+1}^{(n)} = w_{b,i}^{(n)} - \rho g_n^{(b)}$;
Local encryption process;
Initialization:$(pk, sk, ek) \leftarrow FHEkeygen(1^\lambda)$;
noise: $(r_1^{(n)}, r_2^{(n)}, ..r_n^{(n)}) \leftarrow SP$;
for $n \in SP$ **do**
  $[[w^{weight}]] = DPenc(r^{(n)}, w^{weight})$;
  $[[w^{Bias}]] = FHEenc(pk, w^{Bias})$;
  $[[w^{(n)}]] = [[w^{(Weight+Bias)}]]$;

Result: $[[w^{(n)}]](n \in SP)$

c) Parameters Aggregation: Once local nodes have successfully uploaded the trained model parameters to the edge blockchain node, the smart contract will start executing tasks with secure aggregation algorithm, as shown in Algorithm 2. The objective of this algorithm is to perform aggregation computation on the uploaded model parameters, obtaining the global model parameters w_{global} , while ensuring the preservation of data privacy. To mitigate computational resources, we employ hierarchical encryption for the parameters. Specifically, differential privacy encryption is applied to the weight layer, while homomorphic encryption is utilized for the bias layer.

$$[[w_{global}^{weight}]] = \sum_{i \in SP} q_i [[w_i^{weight}]] \quad (4)$$

$$[[w_{global}^{bias}]] = (q_i \otimes [[w_i^{bias}]]) \oplus ...(q_N \otimes [[w_N^{bias}]]) \quad (5)$$

d) Broadcast and Decryption: After undergoing the consensus algorithm, the global model parameters will be transmitted to each local node for decryption. Local nodes utilize their private key sk_i to decrypt the received model parameters and apply the decrypted values to their respective local models. By employing the global model parameters, local nodes can efficiently update and enhance their models without compromising individual data privacy. Decrypted parameters facilitate local nodes in adapting more effectively to industry and user demands' fluctuations.

$$\begin{aligned} w_{global}^{weight} &= [[w_{global^{weight}}]] - \sum_{i=1}^{N} q_i r_i \\ &= (\sum_{i}^{N} q_i w_i^{weight} + \sum_{i}^{N} q_i r_i) - \sum_{i}^{N} q_i r_i \\ &= \sum_{i}^{N} q_i w_{i^{weight}} \end{aligned} \quad (6)$$

$$w_{global}^{bias} = FHEdec(sk_i, [[w_{global}^{bias}]]) \quad (7)$$

Algorithm 2: Secure Aggregation Contracts and local decrypt

Input: $[[w^{(n)}]](n \in SP)$
Output: w_{global}
Data: $(pk, sk, ek), q, SP((P_1, P_2, ...P_N) \in SP), t$

security aggregation process;
$Weight$ $[[w_{global}^{Weight}]] = \sum_{i \in SP} \rho_i [[w_i^{Weight}]]$;
$[[w_{global}^{Bias}]] = (q_1 \bigotimes_{ek} [[w_1^{Bias}]]) \bigoplus ...(q_N \bigotimes_{ek} [[w_N^{Bias}]])$;
$[[w_{global}]] = [[w_{global}^{Weight+Bias}]]$;
for $n \in SP$ **do**
  $w_{global}^{weight} = DPdec(r^{(n)}, [[w_{global}^{weight}]])$;
  $w_{global}^{Bias} = FHEdec(sk, w_{global}^{Bias})$;
  $w_{global}^{(n)} = w^{(Weight+Bias)}$;

Result: w_{global}

During the decryption process, each local node utilizes the traceability of the blockchain to retrieve the previously generated random noise which is used for obtaining the decrypted weight layer. As for the bias layer, each node employs their generated private key for decryption.

e) Iteration: Local nodes continue with the process of local training and uploading, aiming to further enhance the model's performance. This iterative cycle persists until the model meets the application requirements. Through repeated local training and parameter uploading, each local node continually updates and improves its model, allowing it to adapt to evolving data and environmental changes.

In summary, the workflow of the blockchain-based edge federated learning model can be divided into the above steps. This workflow combines the security of blockchain and the advantages of distributed computing, achieving a secure, efficient, and trustworthy model training and updating process. As a result, it provides improved performance and privacy protection for applications.

B. Method Discussion

In the context of smart wearable devices, the combination of FL and DP offers several advantages. FL allows for decentralized training on user devices, addressing privacy

concerns by keeping raw data locally. DP, on the other hand, provides an additional layer of privacy protection by adding noise to the aggregated gradients during the model aggregation process.

By combining FL and DP, we can protect user privacy while obtaining accurate insights from models. This combination allows for meaningful analysis while ensuring sensitive data remains protected. Our experiments showed that using only DP significantly reduced model accuracy. To balance performance and privacy, we chose a hybrid privacy scheme that combines DP and homomorphic encryption (HE). This approach ensures privacy while also maintaining a good balance of accuracy and performance in the model.

C. Convergence Analysis

In this section, we will demonstrate theoretically how the convergence of our improved model is achieved.

Suppose the following edge federation learning scenario: We have a set N ,of edge nodes, and each edge node participating in federated learning possesses its own local dataset D_N,where $D_n = d_{d_{n,1},d_{n,2}...,d_{n,m}}, D$ is the collection of all datasets. The computational power for local model training is provided by the edge nodes. Periodically, each edge node uploads its locally trained model parameters to perform FedAvg aggregation.

The loss function for the dataset D_n owned by each edge node n is shown as follows:

$$F_n = \frac{1}{|D_n|} \sum_{i=1}^{|D_n|} f_n\left(w, d_{n,i}\right) \tag{8}$$

where $f_n()$is the loss function for the $i-th$ data sample of the model parameters w.And the objective function for local training is defined in Eq. 9.

$$F_N^* = minF_n \tag{9}$$

Then the global loss function can be defined as

$$F_n^* = min \sum_{n=1}^{N} q_n F_n(w). \tag{10}$$

where $q_n = \frac{|D_n|}{D}$. We use SGD as the optimization algorithm for the objective function, where the client samples a random and uniform data from its own dataset for training. The number of iterations for local SGD is denoted as t. For every E intervals, the edge node sends its local parameters to the aggregator for FedAvg aggregation. And the aggregated parameters are then redistributed to each node.

The update of model weights during the local training process is defined as follows:

$$v_{t+1}^n = w_t^n - \rho_t \nabla F_n\left(w_t^n, c_t^n\right) \tag{11}$$

where ρ is the learning rate and c represents the randomly selected sample from dataset D_n. If $(t+1)$ is a multiple of E during the local training process, the model parameter aggregation update is shown as follows:

$$w_{t+1}^n = \sum_{n=1}^{\mathrm{N}} q_n v_{t+1}^n. \tag{12}$$

Otherwise,

$$w_{t+1}^n = v_{t+1}^n \tag{13}$$

In the following, we give some assumptions and definitions for convergence proofs.

Assumption 1. *Loss function of all edge nodes $F_n : n = 1, 2...N$ is all L-smooth in Eq. 14 and $\mu - stronglyconvex$ in Eq. 15.*

$$\begin{aligned} \mathrm{F}^{\mathrm{k}}(\boldsymbol{b}) - \mathrm{F}^{\mathrm{k}}(\boldsymbol{a}) \leq \nabla F(\boldsymbol{a})^{\top}(\boldsymbol{b}-\boldsymbol{a}) \\ + \frac{\mathrm{L}}{2}\|\boldsymbol{b}-\boldsymbol{a}\|^2 \end{aligned} \tag{14}$$

$$\begin{aligned} \mathrm{F}^{\mathrm{k}}(\boldsymbol{b}) - \mathrm{F}^{\mathrm{k}}(\boldsymbol{a}) \geq \nabla F(\boldsymbol{a})^{\top}(\boldsymbol{b}-\boldsymbol{a}) \\ + \frac{\mu}{2}\|\boldsymbol{b}-\boldsymbol{a}\|^2. \end{aligned} \tag{15}$$

Assumption 2. *When each edge node performs SGD, the stochastic gradient of c^n is bounded*

$$\mathrm{E}||\nabla F_n\left(w^n, c^n\right)^2|| \leq \mathrm{G}^2. \tag{16}$$

Assumption 3. *The modulus of the stochastic gradient at each edge node is bounded*

$$\mathrm{E}||\nabla F_n\left(w^n, c^n\right)^2|| \leq \mathrm{G}^2. \tag{17}$$

Definition 2. *The weighted sum of gradients for each node in a single SGD step is denoted as*

$$g_t = \sum_{n=1}^{\mathrm{N}} q_n \nabla F_n\left(w_t^n, c_{\mathrm{t}}^n\right). \tag{18}$$

And, weighted gradient at global single aggregation is denoted as

$$\bar{g}_t = \sum_{n=1}^{\mathrm{N}} q_n \nabla F_n\left(w_t\right) = \mathrm{E}\left[g_t\right]. \tag{19}$$

Definition 3. *We define two virtual sequences*

$$\bar{v}_t = \sum_{n=1}^{N} q_n v_t^n, \quad \bar{w}_t = \sum_{n=1}^{N} q_n w_t^n. \tag{20}$$

Then, $\bar{v}_{t+1}$ is obtained from $\bar{w}_t$ by a single step of SGD, denoted as $\bar{v}_{t+1} = \bar{w}_t - \rho_t g_t$.

Lemma 1. *If $\rho_t \leq \frac{1}{4L}$, then the following inequality holds.*

$$\begin{aligned} \mathrm{E}||\overline{\boldsymbol{v}}_{t+1} - \boldsymbol{w}^*||^2 \leq (1-\rho_t\mu)\,\mathrm{E}||\overline{\boldsymbol{w}}_{\mathrm{t}} - \boldsymbol{w}^*||^2 \\ + \rho_t^2 \mathrm{E}||\boldsymbol{g}_t - \overline{\boldsymbol{g}}_t||^2 + 6\ \mathrm{L}\rho_t^2\mathrm{H} \\ + 2\mathrm{E}\sum_{n=1}^{\mathrm{N}} \rho_t ||\overline{\boldsymbol{w}}_t - \boldsymbol{w}_n^t||^2 \end{aligned} \tag{21}$$

Where $\mathrm{H} = \mathrm{F}^* - \sum_{n=1}^{\mathrm{N}} q_n \mathrm{F}_n^* \geq 0$.

Proof.

$$\begin{aligned}\left\|\bar{v}_{t+1}-w^{*}\right\|^{2}=&\left\|\bar{w}_{t}-w^{\hat{v}^{2}}\right\|^{2}-2\rho_{t}\left\langle\bar{w}_{t}-w,\bar{g}_{t}\right\rangle\\&+\rho_{t}^{2}\left\|\bar{g}_{t}\right\|^{2}+\rho_{t}^{2}\left\|g_{t}-\bar{g}_{t}\right\|^{2}\end{aligned}\tag{22}$$

where $2\mathrm{E}||\rho_t\left\langle\bar{w}_t-w^*-\rho_t\bar{g}_t,\bar{g}_t-g_t\right\rangle||^2=0$
Based on Eq. 14, we have

$$\left\|\nabla F_{n}\left(w_{t}^{n}\right)\right\|^{2}\leq 2L\left(F_{n}\left(w_{t}^{n}\right)-F_{n}^{*}\right).\tag{23}$$

Thus, Eq. 24 holds

$$\rho_{t}^{2}\left\|\bar{g}_{t}\right\|^{2}\leq 2L\rho_{t}^{2}\sum_{n=1}^{N}q_{n}\left(F_{n}\left(w_{t}^{n}\right)-F_{n}^{*}\right).\tag{24}$$

Then, Eq. 25 holds

$$\begin{aligned}-2\rho_{t}\left\langle\bar{w}_{t}-w\bar{g}_{t}\right\rangle&=-2\rho_{t}\sum_{n=1}^{N}q_{n}\left\langle\bar{w}_{t}-w,\nabla F_{n}\left(w_{t}^{n}\right)\right\rangle\\&=-2\rho_{t}\sum_{n=1}^{N}q_{n}\left\langle\bar{w}_{t}-w_{t}^{n},\nabla F_{n}\left(w_{t}^{n}\right)\right\rangle\\&\quad-2\rho_{t}\sum_{n=1}^{N}q_{n}\left\langle w_{t}^{n}-w^{\mathring{\mathrm{a}}},\nabla F_{n}\left(w_{t}^{n}\right)\right\rangle.\end{aligned}\tag{25}$$

By using Cauchy-Schwarz inequality and AM-GM inequality, we have

$$\begin{aligned}&-2\left\langle\bar{w}_{t}-w_{t}^{n},\nabla F_{n}\left(w_{t}^{n}\right)\right\rangle\\&\leq\frac{1}{\rho_{t}}\left\|\bar{w}_{t}-w_{t}^{n}\right\|^{2}+\rho_{t}\left\|\nabla F_{n}\left(w_{t}^{n}\right)\right\|^{2}.\end{aligned}\tag{26}$$

According to Eq. 15, we have

$$\begin{aligned}&-\left\langle w_{t}^{n}-w,\nabla F_{n}\left(w_{t}^{n}\right)\right\rangle\\&\leq-\left(F_{n}\left(w_{t}^{n}\right)-F_{n}\left(w^{*}\right)\right)-\frac{\mu}{2}\left\|w_{t}^{n}-w\right\|^{2}.\end{aligned}\tag{27}$$

Thus, we have the following inequality holds.

$$\begin{aligned}\left\|\bar{w}_{t}-w,-\rho_{t}\bar{g}_{t}\right\|^{2}\leq&\left(1-\mu\rho_{t}\right)\left\|\bar{w}_{t}-w^{\mathfrak{a}}\right\|^{2}\\&+\sum_{n=1}^{N}q_{n}\left\|w_{t}-w_{n}^{t}\right\|^{2}\\&-\zeta_{t}\sum_{n=1}^{N}q_{n}\left(F_{n}\left(w_{t}^{n}\right)-F_{n}\left(w^{*}\right)\right)\\&+4L\rho_{t}^{2}\mathrm{H}\end{aligned}\tag{28}$$

Where $\zeta_t=2\rho_t\left(1-2L\rho_t\right),\rho_t\leq\frac{1}{4L}$ and $\rho_t\leq\zeta_t\leq2\rho_t$
By using AM-GM inequality, we have

$$\begin{aligned}&\sum_{n=1}^{N}q_{n}\left(F_{n}\left(w_{t}^{n}\right)-F_{n}\left(w^{*}\right)\right)\\&\geq-\sum_{n=1}^{N}q_{n}\left[\rho_{t}L\left(F_{n}\left(\bar{w}_{t}\right)-F_{n}^{*}\right)+\frac{1}{2\rho_{t}}\left\|w_{t}^{n}-\bar{w}_{t}\right\|^{2}\right]\\&+\left(F\left(\bar{w}_{t}\right)-F^{*}\right).\end{aligned}\tag{29}$$

Thus

$$\begin{aligned}&-\zeta_{t}\sum_{n=1}^{N}q_{n}\left(F_{n}\left(w_{t}^{n}\right)-F_{n}\left(w^{*}\right)\right)+4L\rho_{t}^{2}\mathrm{H}\\&\leq\sum_{n=1}^{N}q_{n}\left\|\bar{w}_{t}-w_{n}^{t}\right\|^{2}+6L\rho_{t}^{2}\mathrm{H}\end{aligned}\tag{30}$$

Therefore, Eq. 23 can be written as

$$\begin{aligned}\left\|\bar{w}_{t}-w-\rho_{t}\bar{g}_{t}\right\|^{2}\leq&\left(1-\mu\rho_{t}\right)\left\|\bar{w}_{t}-w\right\|^{2}\\&+\sum_{n=1}^{N}q_{n}\left\|\bar{w}_{t}-w_{n}^{t}\right\|^{2}+6L\rho_{t}^{2}\mathrm{H}.\end{aligned}\tag{31}$$

According to $\left\|\bar{w}_t-w^*-\rho_t\bar{g}_t\right\|^2=\left\|\bar{w}_t-w^2\right\|^2-2\rho_t\left\langle\bar{w}_t-w,\bar{g}_t\right\rangle+\rho_t^2\left\|\bar{g}_t\right\|^2$, and the expectations on both sides of the equation, we can complete the proof. □ According to Assumption 2, we have

$$\begin{aligned}\mathrm{E}\left\|g_{t}-\bar{g}_{t}\right\|^{2}&=\mathrm{E}\left\|\sum_{n=1}^{N}q_{n}\left(\nabla F_{n}\left(w_{t}^{n},c_{t}^{n}\right)-\nabla F_{n}\left(w_{t}^{n}\right)\right)\right\|^{2}\\&=\sum_{n=1}^{N}q_{n}^{2}\mathrm{E}\left\|\nabla F_{n}\left(w_{t}^{n},c_{t}^{n}\right)-\nabla F_{n}\left(w_{t}^{n}\right)\right\|^{2}\\&\leq\sum_{n=1}^{N}q_{n}^{2}\sigma_{n}^{2}.\end{aligned}\tag{32}$$

According to Assumption 3, if $\rho_i\leq2\rho_{t+E}$, then the following inequality holds.

$$\begin{aligned}\mathrm{E}\sum_{n=1}^{N}q_{n}\left\|\bar{w}_{t}-w_{t}^{n}\right\|^{2}&=\mathrm{E}\sum_{n=1}^{N}q_{n}\left\|\left(w_{t}^{n}-\bar{w}_{t_{0}}\right)-\left(\bar{w}_{t}-\bar{w}_{t_{0}}\right)\right\|^{2}\\&\leq\mathrm{E}\sum_{n=1}^{N}q_{n}\left\|w_{t}^{n}-\bar{w}_{t_{0}}\right\|^{2}\\&\leq\sum_{n=1}^{N}q_{n}\rho_{t_{0}}^{2}(E-1)^{2}G^{2}\\&\leq4\rho_{t}^{2}(E-1)^{2}G^{2}\end{aligned}\tag{33}$$

Next, let $S_t=E||\bar{w}_t-w||^2$, based on Eq. 16, 27 and 28, we have

$$S_{t+1}\leq\left(1-\rho_{\mathrm{t}}\mu\right)S_{t}+\rho_{\mathrm{t}}^{2}\mathrm{M}.\tag{34}$$

where $\mathrm{M}=\sum_{\mathrm{n}=1}^{\mathrm{N}}q_n^2\sigma_n^2+6\mathrm{LH}+8(\mathrm{E}-1)^2\mathrm{G}^2$.
Suppose that $\rho_t=\frac{\varphi}{t+v}$, when $\rho_1=\frac{\varphi}{1+v}\leq\frac{1}{4L},\frac{\varphi}{t+v}<2\frac{\varphi}{t+v+E}$ and $0<\rho_i\mu<1$ based on mathematical induction, we have

$$\begin{aligned}S_{t}&\leq\frac{\max\left\{\frac{v^{2}\ \mathrm{N}}{v\mu-1},(v+1)S_{1}\right\}}{t+v}\\&\leq\frac{\max\left\{\frac{v^{2}\ \mathrm{N}}{v\mu-1},(v+1)S_{1}\right\}}{t+v+1}=S_{t+1}.\end{aligned}\tag{35}$$

According to Eq. 35, it can be inferred that $S(t)$ decreases with increasing t, and $S(t)$ is a non-negative value. Therefore, it can be concluded that (t) converges as t increases.

V. Experimental analysis

In this section, we conduct extensive experiments on two real dataset to evaluate the performance of the proposed model in this paper. Firstly, we introduce the experimental settings. Secondly, we conduct three experiments to represent the advantage of the proposed model, respectively, to handle the following questions.

Q1: What are the advantages of our proposed model in terms of computation and communication overhead for different parameters and different number of concurrency?

Q2: What are the advantages of our proposed model over Local Model and traditional FL model in terms of accuracy and loss value?

Q3: What are the advantages of our proposed model in terms of robustness under different percentages of client failures?

Q4: What are the advantages of our proposed model in terms of time efficiency and privacy preservation?

A. Experimental setting

In our experiments, we use three cloud servers to evaluate the performance of our proposed Safe-FL model. Each cloud server is configured with 1 core (vCPU) 2 GiB, NVIDIA GeForce GTX 1650 SUPER (8G) GPU, and Ubuntu 18.04 64-bit system.

In our experiments, we use three public datasets as shown in Table I. One dataset comes from user behavior data obtained from six activities (walking, walking upstairs, walking downstairs, sitting, standing, and lying down) on a smartphone (Samsung Galaxy S II) worn by 30 volunteers in the age group of 19-48 years old. The dataset consists of 10,000 data, 8,000 for the training dataset and 2,000 for the testing dataset. Another dataset is the real load data we provide is collected by the smart meter terminal. Smart meter terminals are a typical class of edge terminal devices with the ability of encryption, storage and edge computing, and can provide the load data base for power consumption information collection system.

TABLE I
Experimental datasets

Dataset	Type	#Instance	#Instance for train dataset	#Instance for test dataset	Data resource
Data1	Text	10000	8000	2000	https://aistudio.baidu.com/aistudio/datasetdetail/137267/0
Data2	Text	320	200	120	https://github.com/dxdora/data-resourse

To test the performance of our proposed model, we use the CKKS fully homomorphic encryption scheme from the Microsoft SEAL open-source project [43] at the blockchain network layer and add Laplace noise at the model gradient merging layer.Here we use the parameter $poly = 2048$. Then, we obtain the user's local gradient parameters through a local training model and encrypt the local gradient model using a hybrid encryption scheme. Finally, we configure Safe-FL, a secure federation security aggregation algorithm, in the Hyperledger Fabric chain [44] and use the Raft [45] consensus algorithm.

B. Evaluation metrics

1) Computational overhead: In our experiments, we evaluate the computational overhead of our proposed model by testing the key generation time T_{kg}, encryption time T_{En} , decryption time T_{Dc} , transaction processing delay T_{Td}, and security gradient aggregation time T_{Sa} for three cloud servers (users). Also, we evaluate the comprehensive performance of the model in our experiments by calculating the total overhead for datasets with different model parameter sizes at different concurrency numbers as shown in Eq. 36.

$$T_{Total} = T_{Kg} + T_{En} + T_{De} + T_{Td} + T_{Sa} \tag{36}$$

2) Accuracy and Loss rate: In our experiments, we measure the performance of the global model using the accuracy ACC as well as the loss value LOSS.

Let the size of the edge nodes be N and the size of the local dataset corresponding to each edge node i be $|D_i|, i \in [1, N]$, then the total size of the dataset in the local model training is $D = D_1, D_2, ...D_N$, and $|D| = \sum_{i\in[1,N]} D_i$.

The edge user behavior analysis in this paper is considered as a multiclassification problem. Then the accuracy ACC and loss value LOSS are defined as follows, respectively.

Definition 4. *Let $A_i, i \in [1, N]$ be the accuracy of user behavior analysis of the $i-th$ edge node, then $ACC = \frac{\sum_{i=1}^{N} A_i}{|D|}$ is called the accuracy of the global model.*

Definition 5. *Let $F_i = \frac{1}{|D_i|}\sum_{i=1}^{|D_i|} f_i(w)$ be the loss value of user behavior analysis of the i-th edge node, then $LOSS = \sum_{i=1}^{N} \frac{|D_i|}{D|} F_n(w)$ is called the loss value of the global model.*

Definition 6. *Let $PCCs = \frac{Cov(X,Y)}{\sigma(X)\sigma(Y)}$ be the Pearson correlation coefficient between the original gradient and the safety gradient.*

In this paper, we use loss value to analyze convergence of Safe-FL. The smaller the loss value, the better the model convergence.

C. Experimental analysis

1) Computational overhead (Q1): In our scheme, $N = 10$ client components are used to form a training model network and each edge node is considered as a blockchain node.The initial learning rate is $\eta = 0.005$,$S = 5$ represents the number of iterations after which a local aggregation is performed, and it needs to be adaptively decreased as the iterations progress.. With this design approach, we can build a blockchain network with a large scale and where each client can contribute computational and storage resources. In a real blockchain network, the number of transactions processed simultaneously is referred to as the concurrency level. In our experiments, we compare the computational overhead at different concurrency levels, as shown in Fig.3, Fig.4 and Fig.5, respectively.

For Data1 and Data2, by analyzing these figures, we can draw the following conclusions:

- For the same model parameter size, different user concurrency level does not have any effect on the key generation time, encryption time and decryption time.

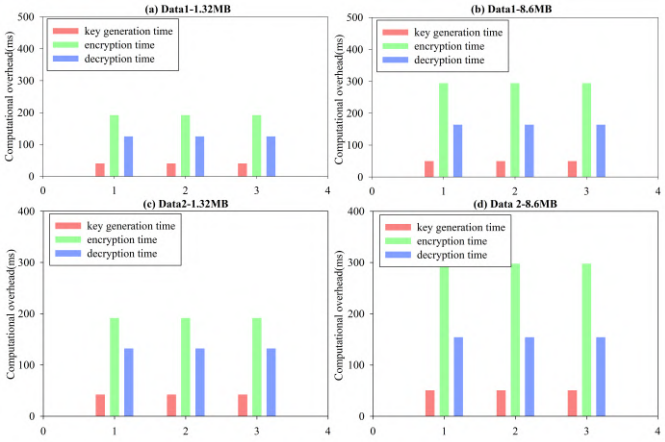

Fig. 3. Key generation time, encryption time and decryption time for experimental datasets with different model parameter sizes at different concurrency levels.

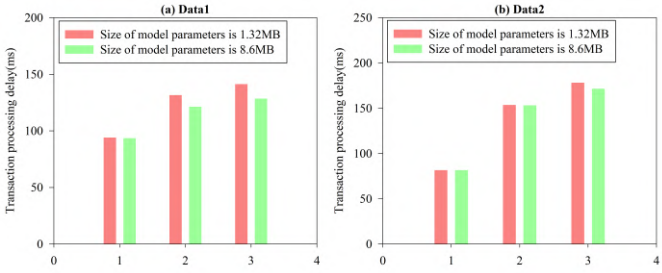

Fig. 4. Transaction processing delay for experimental datasets with different model parameter sizes at different concurrency levels.

And, encryption time is greater than the key generation time and decryption time for different model parameter sizes and user concurrency levels.

- For the same user concurrency levels, the larger the model parameter size, the larger the key generation time, encryption time, and decryption time.
- Under the same model parameter size, the higher user concurrency levels, the larger the transaction delay and security gradient aggregation time. Under the same user concurrency level, the larger the model parameter size, the lower the transaction delay and the higher security gradient aggregation time.

From Fig.6, it can be seen that the total computational overhead increases with the increasing user concurrency level under the same model parameter size. According to Eq. 36, the percentage of encryption time in the total computational overhead is the largest regardless of the model parameter size and user concurrency level.

During the iteration process, the encryption and decryption time locally at the client occupies the largest computational overhead. However, considering the privacy security protection mechanism provided by Safe-FL, we believe that this computational overhead is acceptable at about 20% of the total overhead. Overall, although Safe-FL is designed to add some transaction processing delay, it provides a strong privacy security protection mechanism.

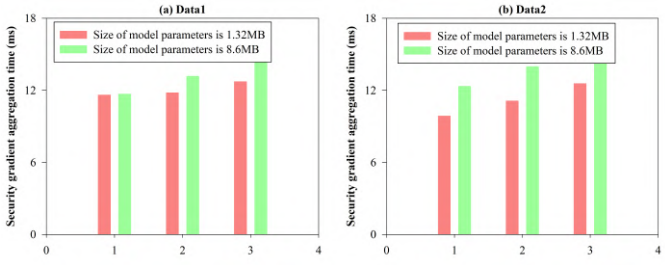

Fig. 5. Security gradient aggregation time for experimental datasets with different model parameter sizes at different concurrency levels.

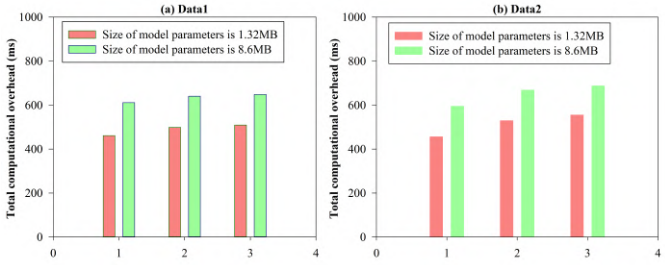

Fig. 6. Total computational overhead for different datasets with different model parameters and user concurrent levels.

2) Accuracy and loss value (Q2): For all experimental datasets, this experiment compares the advantages of the Safe-FL model in different training modes in terms of model accuracy and loss values compared to Local Model and traditional federated learning model (FL). The experimental results of accuracy and loss values between different models are shown in Fig. 7 and Fig. 8, respectively.

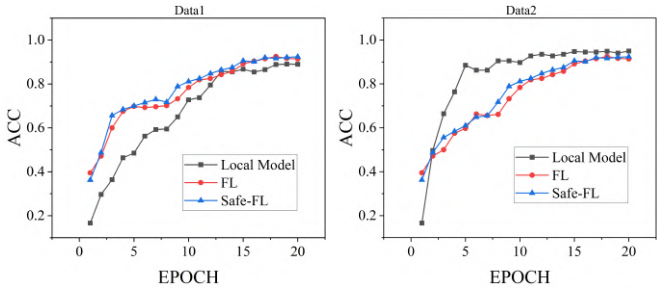

Fig. 7. Comparison of model accuracy of experimental datasets in different training modes.

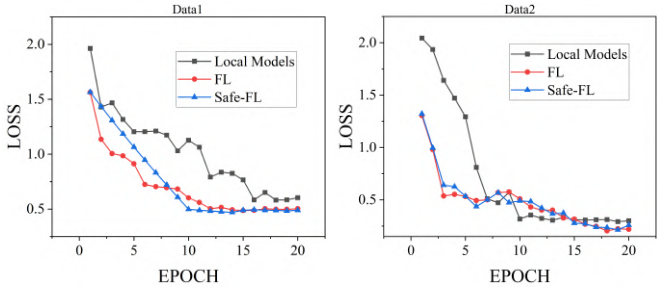

Fig. 8. Comparison of model loss value of experimental datasets in different training modes.

From Fig.7, we can get the following results.

- For the two experimental datasets, the accuracy of our proposed Safe-FL model gradually increases as the number of iterations increases until it finally converges to 0.92375 and 0.7995, respectively. Especially, for Data2, when the number of iterations increases to 15, the accuracy of the Safe-FL model is almost the same as the FL model.
- Our proposed Safe-FL model protects the privacy of participants in the edge user behavior analysis by introducing secure data aggregation and encryption techniques into the FL model, while ensuring the accuracy of the model.

From Fig.8, we can have the following conclusions.

- For Data1 and Data2, our proposed Safe-FL model converges faster than Local Model and FL model. For edge user behavior analysis, the faster the model converges, the less resources it consumes, which is more conducive to practical deployment for use in edge networks.
- The Safe-FL model also enables it to ensure that the private data of each edge participant is protected from disclosure or misuse during the federated learning process by employing privacy-preserving techniques and secure data aggregation algorithms.

The experimental results demonstrate that the privacy protection measures do not significantly affect the convergence speed of the model. Thus, our proposed model can guarantee data privacy while still achieving convergence performance comparable to that of traditional FL models.

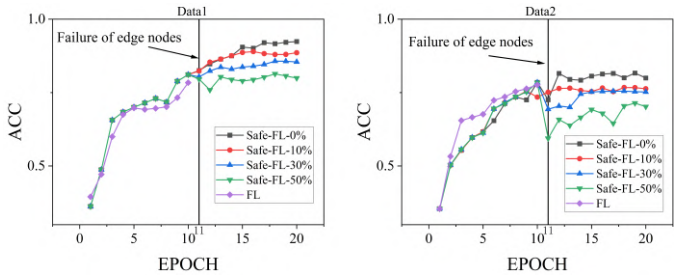

Fig. 9. Comparison of model accuracy for edge node failures during training.

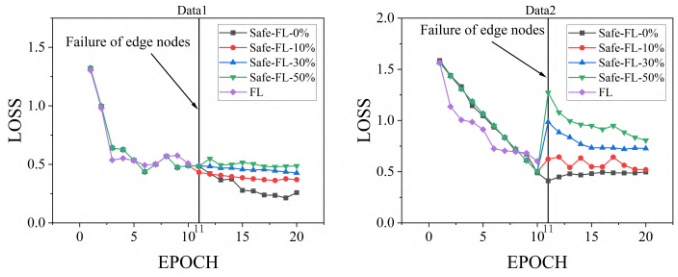

Fig. 10. Comparison of model loss value for edge node failures during training.

3) Robustness of model (Q3): In order to test the robustness of the model, this experiment is iterated for 20 times and compares the accuracy (ACC) and loss value (LOSS) of our proposed Safe-FL model with other models for different edge user node failure ratios. In the experiments, we have set up $N = 20$ nodes and taken edge node failures of $\alpha = 10\%, 30\%, , 50\%$, respectively. For all experimental datasets, Fig. 9 and Fig. 10 compare the accuracy and loss values of various models when the edge node fails during training, respectively.

From Fig.9 and Fig.10, we can find the following results.

- An edge node failure occurs when the training reaches the 11th generation and the FL model fails. This causes its accuracy and loss value to be zero. However, the accuracy and loss value of our proposed Safe-FL model remain unaffected.
- The accuracy of Safe-FL model is lower and the loss value of it is higher when the edge node failure ratio is higher. Nevertheless, for Data1 and Data2, when the edge node failure ratio reaches 30% and 50%, the highest accuracy of Safe-FL model still reaches 0.8545, 0.7994; 0.7542, 0.7082, respectively.
- For Data1 and Data2, the accuracy of the Safe-FL-0% model is improved by about 4.12%, 7.502%, and 13.47%; 4.89%, 5.66%, and 11.41%, respectively, compared to the Safe-FL-10%, Safe-FL-30%, and Safe-FL-50% models.
- For Data1 and Data2, compared with the Safe-FL-10%, Safe-FL-30%, and Safe-FL-50% models, the loss values of Safe-FL-0% model are reduced by about 32.17%, 41.23%, and 48.41%; 5.64%, 32.61%, and 39.22%, respectively.

The experimental results show that our proposed Safe-FL model still has high accuracy and low loss values under the conditions of edge node failure. It also demonstrates that the Safe-FL model has better robustness and reliability in the face of edge node user failures, and is able to effectively handle model training under incomplete or anomalous data.

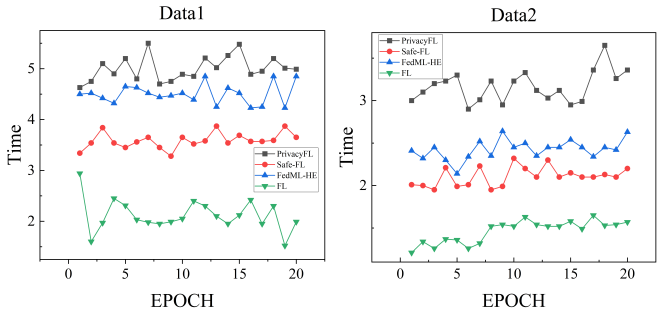

Fig. 11. Comparison of model communication time value of experimental datasets in different training modes.

4) Time efficiency and privacy preservation of model (Q4): In order to measure the time efficiency as well as the privacy preservation of Safe-FL, the experiment measured the time spent on each EPOCH. The Pearson coefficient [46] was used to measure the correlation between the safe gradient and the original gradient, which in turn responds to the model's gradient privacy protection during communication.The Pearson's coefficient takes values between -1-1, with closer to 0 indicating a weaker correlation.

From Fig.11 we can find the following results.

- From the Fig.11, we can observe that our model has an average overhead of approximately 3.5 seconds, indicating a significant efficiency improvement compared to the similar

PrivacyFL [47] and FedML-HE [48] model. Additionally, the use of blockchain as a replacement for third-party servers has greatly enhanced the security of aggregation.

The communication overhead of Safe-FL model in each EPOCH includes encryption time, encryption algorithm parameter generation time, and blockchain communication delay. According to the data shown in the graph, compared with PrivacyFL and FedML-HE, we have made significant improvements in communication security and overhead by using blockchain layer. Although we have additional encryption and decryption communication overhead compared with the classic FL model, this is inevitable for security reasons.

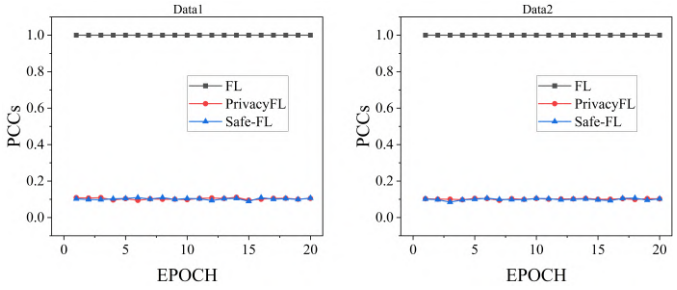

Fig. 12. Gradient and safety gradient PCCs.

From Fig.12 we can find the following results.

- From the Fig.12, We can observe that the PCCs of the security gradient with respect to the original gradient for our model during each EPOCH communication are basically concluded to be uncorrelated between approximately 0.1-0.2, similar to the privacy-preserving performance of PrivacyFL.

The results show that the privacy of the gradient is effectively protected after Safe-FL adopts the secure gradient algorithm. The encryption process successfully destroys the correlation of the original gradient, ensures the security of the gradient data, and effectively prevents the risk of gradient leakage.

VI. Conclusions

To protect user privacy in edge user behavior analysis, this paper first introduces federated learning and blockchain technology to construct a framework for edge user behavior analysis considering privacy protection (Safe-FL). In Safe-FL, first, local model training and encryption algorithms are proposed, and the encrypted data are uploaded to the edge blockchain nodes. Second, a model aggregation algorithm based on secure multiparty computation is proposed to protect edge data privacy. Finally, we theoretically analyze the convergence of Safe-FL. Comparative experimental results on two public datasets show that 1) Safe-FL model has acceptable computational overhead under the condition of protecting edge user privacy; 2) Compared with Local Model as well as FL model, Safe-FL protects edge user privacy while having high accuracy and convergence speed; 3) Compared with Safe-FL-10%, Safe-FL-30%, Safe-FL-50% and FL models, Safe-FL has strong robustness, which can protect edge user privacy as well as maintain high accuracy and convergence speed in the case of partial edge node failure. In future work, the reduction of communication overhead and ciphertext transmission as well as the improvement of network throughput deserve our attention.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (No.51977113, 62293500, 62293501 and 62293505).

References

[1] B. Li, Q. He, G. Cui, X. Xia, F. Chen, H. Jin, and Y. Yang, "Read: Robustness-oriented edge application deployment in edge computing environment," *IEEE Transactions on Services Computing*, vol. 15, no. 3, pp. 1746–1759, 2020.

[2] K. Cao, Y. Liu, G. Meng, and Q. Sun, "An overview on edge computing research," *IEEE access*, vol. 8, pp. 85 714–85 728, 2020.

[3] P. Dai, Y. Huang, X. Wu, K. Li, H. Xing, and K. Liu, "Freshness and security-aware cache update in blockchain-based vehicular edge networks," *IEEE Transactions on Consumer Electronics*, vol. 70, no. 1, pp. 108–121, 2024.

[4] M. Wazid, A. K. Das, and S. Shetty, "Bsfr-sh: Blockchain-enabled security framework against ransomware attacks for smart healthcare," *IEEE Transactions on Consumer Electronics*, vol. 69, no. 1, pp. 18–28, 2022.

[5] A. M. Groba, P. J. Lobo, and M. Chavarrías, "Slack-time closed-loop control system for multimedia mobile devices," *IEEE Transactions on Consumer Electronics*, vol. 64, no. 2, pp. 162–170, 2018.

[6] L. Rachakonda, A. K. Bapatla, S. P. Mohanty, and E. Kougianos, "Sayopillow: Blockchain-integrated privacy-assured iomt framework for stress management considering sleeping habits," *IEEE Transactions on Consumer Electronics*, vol. 67, no. 1, pp. 20–29, 2020.

[7] Y. Zhao, J. Zhao, M. Yang, T. Wang, N. Wang, L. Lyu, D. Niyato, and K.-Y. Lam, "Local differential privacy-based federated learning for internet of things," *IEEE Internet of Things Journal*, vol. 8, no. 11, pp. 8836–8853, 2020.

[8] A. Alzahrani and M. Z. Asghar, "Maintaining user security in consumer electronics-based online recommender systems using federated learning," *IEEE Transactions on Consumer Electronics*, vol. 70, no. 1, pp. 2657–2665, 2024.

[9] J. W. Kim, J. H. Lim, S. M. Moon, and B. Jang, "Collecting health lifelog data from smartwatch users in a privacy-preserving manner," *IEEE Transactions on Consumer Electronics*, vol. 65, no. 3, pp. 369–378, 2019.

[10] M. Yahuza, M. Y. I. B. Idris, A. W. B. A. Wahab, A. T. Ho, S. Khan, S. N. B. Musa, and A. Z. B. Taha, "Systematic review on security and privacy requirements in edge computing: State of the art and future research opportunities," *IEEE Access*, vol. 8, pp. 76 541–76 567, 2020.

[11] W. Wang, F. H. Memon, Z. Lian, Z. Yin, T. R. Gadekallu, Q.-V. Pham, K. Dev, and C. Su, "Secure-enhanced federated learning for ai-empowered electric vehicle energy prediction," *IEEE Consumer Electronics Magazine*, vol. 12, no. 2, pp. 27–34, 2021.

[12] V. P. Yanambaka, S. P. Mohanty, E. Kougianos, and D. Puthal, "Pmsec: Physical unclonable function-based robust and lightweight authentication in the internet of medical things," *IEEE Transactions on Consumer Electronics*, vol. 65, no. 3, pp. 388–397, 2019.

[13] K. Akherfi, M. Gerndt, and H. Harroud, "Mobile cloud computing for computation offloading: Issues and challenges," *Applied computing and informatics*, vol. 14, no. 1, pp. 1–16, 2018.

[14] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, "Adaptive federated learning in resource constrained edge computing systems," *IEEE journal on selected areas in communications*, vol. 37, no. 6, pp. 1205–1221, 2019.

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

[16] M. Nasr, R. Shokri, and A. Houmansadr, "Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning," in *2019 IEEE symposium on security and privacy (SP)*. IEEE, 2019, pp. 739–753.

[17] J. Zhu, J. Cao, D. Saxena, S. Jiang, and H. Ferradi, "Blockchain-empowered federated learning: Challenges, solutions, and future directions," *ACM Computing Surveys*, vol. 55, no. 11, pp. 1–31, 2023.

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

[19] H. Chen and Y. Xiao, "Research on the analysis of users' behavior based on big data," in *2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*. IEEE, 2021, pp. 184–187.

[20] J. Dai, H. Yin, and P. Zhang, "A user behavior clustering algorithm combines association rules and multi-valued discrete features," in *2020 IEEE 5th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA)*. IEEE, 2020, pp. 23–27.

[21] X. Bingbing, Z. Jitao, and W. Xianyi, "A prediction model of user buying behavior based on lstm and svm," in *2021 6th International Symposium on Computer and Information Processing Technology (ISCIPT)*. IEEE, 2021, pp. 26–31.

[22] X. Hu, Y. Yang, S. Zhu, and L. Chen, "Research on a hybrid prediction model for purchase behavior based on logistic regression and support vector machine," in *2020 3rd international conference on artificial intelligence and big data (ICAIBD)*. IEEE, 2020, pp. 200–204.

[23] M. Gan and K. Xiao, "R-rnn: Extracting user recent behavior sequence for click-through rate prediction," *IEEE Access*, vol. 7, pp. 111 767–111 777, 2019.

[24] C. He, G. Liu, S. Guo, and Y. Yang, "Privacy-preserving and low-latency federated learning in edge computing," *IEEE Internet of Things Journal*, vol. 9, no. 20, pp. 20 149–20 159, 2022.

[25] H. Yu, E. Xinhua, X. Li, K. Wang, and S. Zhang, "Multi-behavior recommendation based on simplified graph convolutional networks," in *2021 4th International Conference on Artificial Intelligence and Big Data (ICAIBD)*. IEEE, 2021, pp. 277–282.

[26] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konečnỳ, S. Mazzocchi, B. McMahan *et al.*, "Towards federated learning at scale: System design," *Proceedings of machine learning and systems*, vol. 1, pp. 374–388, 2019.

[27] A. Taïk and S. Cherkaoui, "Electrical load forecasting using edge computing and federated learning," in *ICC 2020-2020 IEEE international conference on communications (ICC)*. IEEE, 2020, pp. 1–6.

[28] S. Liu, J. Yu, X. Deng, and S. Wan, "Fedcpf: An efficient-communication federated learning approach for vehicular edge computing in 6g communication networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 2, pp. 1616–1629, 2021.

[29] Y. Ye, S. Li, F. Liu, Y. Tang, and W. Hu, "Edgefed: Optimized federated learning based on edge computing," *IEEE Access*, vol. 8, pp. 209 191–209 198, 2020.

[30] Y. Qian, L. Hu, J. Chen, X. Guan, M. M. Hassan, and A. Alelaiwi, "Privacy-aware service placement for mobile edge computing via federated learning," *Information Sciences*, vol. 505, pp. 562–570, 2019.

[31] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Differentially private asynchronous federated learning for mobile edge computing in urban informatics," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 3, pp. 2134–2143, 2019.

[32] H. Zhou, G. Yang, Y. Huang, H. Dai, and Y. Xiang, "Privacy-preserving and verifiable federated learning framework for edge computing," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 565–580, 2022.

[33] B. Jia, X. Zhang, J. Liu, Y. Zhang, K. Huang, and Y. Liang, "Blockchain-enabled federated learning data protection aggregation scheme with differential privacy and homomorphic encryption in iiot," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 6, pp. 4049–4058, 2021.

[34] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, and M. Wang, "Lightgcn: Simplifying and powering graph convolution network for recommendation," in *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, 2020, pp. 639–648.

[35] M. Alkhelaiwi, W. Boulila, J. Ahmad, A. Koubaa, and M. Driss, "An efficient approach based on privacy-preserving deep learning for satellite image classification," *Remote Sensing*, vol. 13, no. 11, p. 2221, 2021.

[36] A. R. Javed, M. A. Hassan, F. Shahzad, W. Ahmed, S. Singh, T. Baker, and T. R. Gadekallu, "Integration of blockchain technology and federated learning in vehicular (iot) networks: A comprehensive survey," *Sensors*, vol. 22, no. 12, p. 4394, 2022.

[37] S. Singh, S. Rathore, O. Alfarraj, A. Tolba, and B. Yoon, "A framework for privacy-preservation of iot healthcare data using federated learning and blockchain technology," *Future Generation Computer Systems*, vol. 129, pp. 380–388, 2022.

[38] M. Ali, H. Karimipour, and M. Tariq, "Integration of blockchain and federated learning for internet of things: Recent advances and future challenges," *Computers & Security*, vol. 108, p. 102355, 2021.

[39] S. Banabilah, M. Aloqaily, E. Alsayed, N. Malik, and Y. Jararweh, "Federated learning review: Fundamentals, enabling technologies, and future applications," *Information processing & management*, vol. 59, no. 6, p. 103061, 2022.

[40] K. Munjal and R. Bhatia, "A systematic review of homomorphic encryption and its contributions in healthcare industry," *Complex & Intelligent Systems*, pp. 1–28, 2022.

[41] Y. Zhao and J. Chen, "A survey on differential privacy for unstructured data content," *ACM Computing Surveys (CSUR)*, vol. 54, no. 10s, pp. 1–28, 2022.

[42] T. R. Gadekallu, T. Huynh-The, W. Wang, G. Yenduri, P. Ranaweera, Q.-V. Pham, D. B. da Costa, and M. Liyanage, "Blockchain for the metaverse: A review," *arXiv preprint arXiv:2203.09738*, 2022.

[43] F. Aydin, E. Karabulut, S. Potluri, E. Alkim, and A. Aysu, "Reveal: Single-trace side-channel leakage of the seal homomorphic encryption library," in *2022 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2022, pp. 1527–1532.

[44] D. Ravi, S. Ramachandran, R. Vignesh, V. R. Falmari, and M. Brindha, "Privacy preserving transparent supply chain management through hyperledger fabric," *Blockchain: Research and Applications*, vol. 3, no. 2, p. 100072, 2022.

[45] L. Hou, X. Xu, K. Zheng, and X. Wang, "An intelligent transaction migration scheme for raft-based private blockchain in internet of things applications," *IEEE Communications Letters*, vol. 25, no. 8, pp. 2753–2757, 2021.

[46] X. Liu, H. Li, G. Xu, Z. Chen, X. Huang, and R. Lu, "Privacy-enhanced federated learning against poisoning adversaries," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 4574–4588, 2021.

[47] V. Mugunthan, A. Peraire-Bueno, and L. Kagal, "Privacyfl: A simulator for privacy-preserving and secure federated learning," in *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 2020, pp. 3085–3092.

[48] W. Jin, Y. Yao, S. Han, C. Joe-Wong, S. Ravi, S. Avestimehr, and C. He, "Fedml-he: An efficient homomorphic-encryption-based privacy-preserving federated learning system," *arXiv preprint arXiv:2303.10837*, 2023.

Song Deng (M'16) received the Ph.D.degree in information network from Nanjing University of Posts and Telecommunication, Nanjing, China, in 2009. From 2009 to 2012, he was a Research Fellow with the State Grid Electric Power Research Institute, Nanjing, China. From 2012 to 2014, he was a Research Fellow with the China Electric Power Research Institute, Beijing, China. He is currently the full Professor of Nanjing University of Posts and Telecommunication, Nanjing, China. He was an international visitor with computer science from the University of Louisiana at Lafayette, USA, from 2018 to 2019. His research interests include data security, information security of cyber-physical systems, data mining and knowledge engineering.

Jie Zhang received the B.E. degree in information security from Huaibei Normal University, Huaibei,China in 2021. He is currently pursuing the M.S.degree in computer science at Nanjing University of Posts and Telecommunications, Nanjing, China. His research interests include research on privacy computing and data privacy protection under new power system.

Longxiang Zhang received the B.E. degree in automation from Changzhou University, China in 2022. He is currently pursuing the M.S. degree in electronic information at Nanjing University of Posts and Telecommunications. His research interests include power grid risk assessment and privacy protection.