Hindawi
Mobile Information Systems
Volume 2022, Article ID 2311264, 10 pages
https://doi.org/10.1155/2022/2311264

Research Article

Indoor Positioning Privacy Protection Method Based on Federated Learning in MEC Environment

Jie Wang,[1] **Li Tian**,[1] **Guowei Zhu**,[2] **Chang Liu**,[1] **and Feng Long**[1]

[1]*State Grid Hubei Electric Power Research Institute, Wuhan, Hubei 430077, China*
[2]*State Grid Hubei Electric Power Co. Ltd, Wuhan, Hubei 430077, China*

Correspondence should be addressed to Jie Wang; wangjie.sgcc@alumni.hust.edu.cn

Received 15 July 2022; Revised 17 August 2022; Accepted 30 August 2022; Published 12 October 2022

Academic Editor: Le Sun

Under the current background, it is very important to study the key technologies of new power system edge-to-side security protection for massive heterogeneous power IoT terminals and edge IoT agents, including defense technologies at the levels of device ontology security, communication interaction security, and secure access. *Meaning*. The new power system edge-to-side security protection technology has a summary impact on the privacy protection of indoor positioning. This paper proposes an indoor positioning privacy protection method based on federated learning in Mobile Edge. *Computing (MEC) environment*. Firstly, we analyze the learning mechanisms of horizontal, vertical, and transfer-federated learning, respectively, and mathematically describe it based on the applicability of horizontal and vertical-federated learning under different sample data characteristics. Then, the risk of data leakage when data are used for research or analysis is greatly reduced by introducing differential privacy. In addition, considering the positioning performance, privacy protection, and resource overhead, we further propose an indoor positioning privacy protection model based on federated learning and corresponding algorithms in MEC environment. Finally, through simulation experiments, the proposed algorithm and other three algorithms are, respectively, compared and analyzed in the case of two identical datasets. The experimental results show that the convergence speed, localization time consumption, and localization accuracy of the proposed algorithm are all optimal. Moreover, its final positioning accuracy is about 94%, the average positioning time is 250 ms, and the performance is better than the other three comparison algorithms.

1. Introduction

Indoor positioning refers to the realization of positioning in the indoor environment, mainly using wireless communication, base station positioning, inertial navigation positioning, motion capture, and other technologies to integrate to form a set of indoor position positioning system, so as to realize the positioning of personnel and equipment in indoor space. Precise location monitoring is also important. In order to solve the problem of weak indoor positioning signal and to ensure the positioning accuracy in the practical application environment, the authors in the literature [1] proposed a high-precision positioning algorithm compatible with the two positioning modes through the research on the principles of the algorithms of satellite positioning and ultra wide band (UWB) positioning modes. Under this algorithm, seamless switching between the two positioning modes can be achieved. The high-precision positioning algorithm is compatible with two positioning modes, and the seamless switching between the two positioning modes can be realized under this algorithm. Reference [2] designed an integrated seamless positioning system combining UWB and GNSS for the urgent needs of indoor and outdoor integrated seamless positioning in application scenarios such as power BeiDou security applications and automatic driving navigation. Aiming at the potential defects of traditional RSSI and Taylor series expansion positioning algorithms, literature [3] proposes a set of improved positioning algorithms suitable for power production environments to serve diverse indoor positioning application requirements under smart grids.

Specifically, by introducing processing stages such as Gaussian screening, wavelet transformation, and correcting Taylor series expansion, redundant noise can be fully removed, calculation results can be optimized, and positioning efficiency can be improved.

The rapid development of wireless communication technology in new power systems has greatly promoted the widespread popularization and application of smart power IoT terminals, and location information is essential in the related services of these smart terminal devices [4]. By collecting specific location information, the intelligent terminal realizes location query and acquisition services, path planning and navigation services, target object recognition services, and location sending and query services in emergency situations [5, 6]. Location-based services (LBS) have been widely used in various fields including smart grids. While LBS brings convenience to people's lives, it is also accompanied by the risk of location privacy leakage. When users enjoy indoor location services, they often have concerns about their privacy and security [7–9].

In addition, the rapid growth in the number of users and smart terminal devices has led to an exponential growth of data at the edge of network. Thus, the traditional method of centrally storing data in cloud computing centers for data processing has become increasingly infeasible [10–12]. Mobile edge computing (MEC) technology provides an effective solution to the storage and processing of massive data at the edge of mobile networks. It effectively solves the huge pressure problem brought by massive data transmission and storage to network and cloud storage center by sinking data storage and computing to network edge [13, 14]. However, as a distributed data processing method, MEC nodes may involve the mutual exchange of sensitive data in the process of cooperative data processing, which may lead to data privacy leakage [15].

At present, under the edge computing architecture, the location model can be trained by collecting and sending wireless signal strength information to the cloud through intelligent edge devices, but how to protect some records with sensitive information of electricity users (such as location information, electricity operation, travel arrangements, and work and rest time), and avoid using these sensitive information to infer other privacy information that users do not want to disclose, such as electricity preferences, home furnishings, consumption levels, behavioral habits and social relations, etc. are the problems that need to be solved urgently at present [16, 17].

In order to solve the problems of low positioning accuracy, long time-consumption, and difficulty to resist cross-attack in traditional indoor positioning privacy protection methods, this paper proposes an indoor positioning privacy protection method based on federated learning in MEC environment. Compared with the traditional indoor positioning privacy protection method, the innovation of proposed method is as follows:

(1) The possibility of user privacy data leakage is greatly reduced, and the risk of data destruction is eliminated by introducing differential privacy

(2) The corresponding model is constructed by comprehensively considering the positioning performance, privacy protection, and resource overhead, which improves the comprehensive performance of the privacy protection algorithm

The remaining chapters of this paper are arranged as follows: the second chapter introduces the relevant research in this field; the third chapter introduces the privacy protection method based on federated learning; the fourth chapter is the experimental part, which verifies the performance of the proposed method; and the fifth chapter summarizes the research.

2. Related Work

For the privacy protection method of user indoor positioning in MEC environment, scholars have done related research and achieved certain research results.

Reference [18] proposed a user location privacy protection algorithm based on the improved k-means algorithm and l-diversity idea using the Laplacian mechanism. They proposed a user query privacy protection algorithm based on the k-anonymity algorithm and then proposed a differential privacy-based LBS privacy protection scheme. However, this method did not consider the privacy issues of data collection and aggregation operations and has a limited scope of application. Reference [19] divided the entire fingerprint database based on the E-M clustering algorithm. On this basis, a privacy protection scheme is proposed in Wi-Fi fingerprint-based positioning PPWFL using Wi-Fi devices, but this method cannot use corresponding indicators to evaluate high user privacy. Reference [20] proposed an RPL algorithm that can divide the privacy level of sensitive road segments based on the topology relationship of the road network. On this basis, the differential privacy location protection mechanism DPLPM was used to allocate privacy budget for sensitive road sections to realize the privacy protection of location data. However, this method can only work in completely trusted indoor scenes, and it was difficult to resist differential attacks in untrusted situations. Reference [21] proposed a fingerprint-based high-precision indoor positioning system with positioning accuracy, operating ability in changing nonline-of-sight environments, and computational simplicity as objective functions, and realized user privacy by using multipath propagation to disguise the user's location. However, this method cannot meet the high computational requirements of user privacy protection. Aiming at the balance between geographic location protection and semantic location protection, reference [22] proposed an optimized privacy differential privacy scheme with reinforcement learning in vehicular ad hoc networks based on differential privacy. However, this method led to low localization accuracy due to destruction of the original data distribution. Reference [23] analyzed the reasons for leakage of mobile user location privacy and the deficiencies of existing privacy protection technologies in 5G environment. Combining the preliminary processing of dimensionality reduction, fusion privacy algorithm, and transmission encryption method, a fusion positioning privacy protection method suitable for 5G environment was proposed. However, this method cannot achieve robustness to the changing environment of moving objects. Reference [24] proposed edge crowdsourcing indoor localization architecture to address the problem of privacy leakage in the development of large-scale indoor localization systems. Besides, a privacy-aware indoor positioning algorithm based on secure multiparty computation was presented to protect location privacy. However, the positioning time of this method was long and the efficiency was low.

3. Privacy Protection Method Based on Federated Learning

3.1. Federated Learning Mechanism. Traditional machine learning methods are facing two major challenges: data silos and data security and privacy. Federated learning has been proposed as a possible solution due to its ability to provide a learning protocol for collaboration and security. As a new modeling mechanism, federated learning can uniformly model data from multiple parties without compromising data privacy and security. That is, many clients jointly train the same model under the coordination of central server and cannot disclose their respective data and keep the training data decentralized. It can be better applied to fields where data cannot be directly aggregated for training machine learning models due to factors such as intellectual property rights, privacy protection, and data security.

For the case that multiple data owners want to combine the data, they have to train a machine learning model, and the traditional method is to integrate all data together and to use the integrated data for training to obtain the final model M_s. However, this scheme is usually difficult to implement due to legal issues such as privacy and data security. However, federated learning can solve this problem very well. Federated learning is the ability to obtain a model M_r through training without the data owner having to disclose its own data. Federated learning can ensure that the performance gap between model M_s and model M_r is small enough. When the following formula (1) is established, it means that the accuracy loss of the federated learning algorithm is λ:

$$\left|X_s - X_r\right| < \lambda. \tag{1}$$

In formula (1), X_s represents the performance of model X_s. X_r represents the performance of model X_r. λ is a nonnegative real number.

The data owned by k data owner is represented by a matrix D_k. In matrix D_k, each row represents a sample, and each column represents a feature. The feature space of data is denoted by T. In addition, some data of users may contain specific labels, such as customer value in the field of electricity, health in the field of medical care, and purchasing power in the field of sales. The label space is represented by L. The ID space of the sample is denoted by I. The feature space T, the label space L, and ID space I constitute the complete training dataset (I, T, L). Since the feature space and sample space of data subject may not be the same, it is necessary to distribute the data according to the location of data among the parties in the feature and sample ID space. The federated learning is divided into horizontal-federated learning, vertical-federated learning, and transfer-federated learning. Different types of federated learning architectures are shown in Figure 1.

Different types of federated learning architectures are used in different situations. When there is a large overlap of dataset features and a small sample overlap between participants, horizontal-federated learning is used. When the feature overlap is small and the sample overlap is large, vertical-federated learning is used. Transfer-federated learning is used when both features and samples overlap less. This paper mainly studies horizontal-federated learning and vertical-federated learning.

In the dataset of federated learning participants, when the same sample ID is few but the same data features are many, we often use the horizontal-federated learning technology. For example, for two telecom operators with the same business but different users, the datasets they generate have less overlapping in sample IDs but larger feature overlaps. If we need to combine the data of both parties to build a machine learning model, we often do not copy data directly due to data privacy issues. At this point, we can use the horizontal-federated learning method to model the data. Horizontal-federated learning is to combine samples with the same characteristics from multiple participants for modeling. We take out the part of data with different sample IDs and the same characteristics to train the model. In the horizontal-federated learning solution proposed by Google for Android mobile phone model update, users using Android mobile phones can update the parameters of the model locally and upload the parameters of the model to the Android cloud server. Other data owners can collaborate with the user to train a centralized model.

Horizontal-federated learning is also known as feature-aligned federated learning; that is, the features of the participants' data samples are the same. The word "horizontal" in horizontal-federated learning can be understood as "horizontal division." The training samples of each participant can be regarded as horizontally divided from the total samples. Horizontal-federated learning expands the number of training samples. Therefore, horizontal-federated learning can be summarized by

$$T_a = T_b, L_a = L_b, I_a \neq I_{b,} \forall D_a, D_b\, a \neq b. \tag{2}$$

Vertical-federated learning can also be called feature-based federated learning. It is suitable for cases where there are many overlapping sample IDs in multiple model training participant datasets but few common features. Vertical-federated learning is to combine samples with the same ID from multiple participants for model training. The dataset of each party is divided vertically (that is, divided according to the feature dimension), and the part of data with the same sample ID and different characteristics is taken out to train the model. For example, in two different organizations with

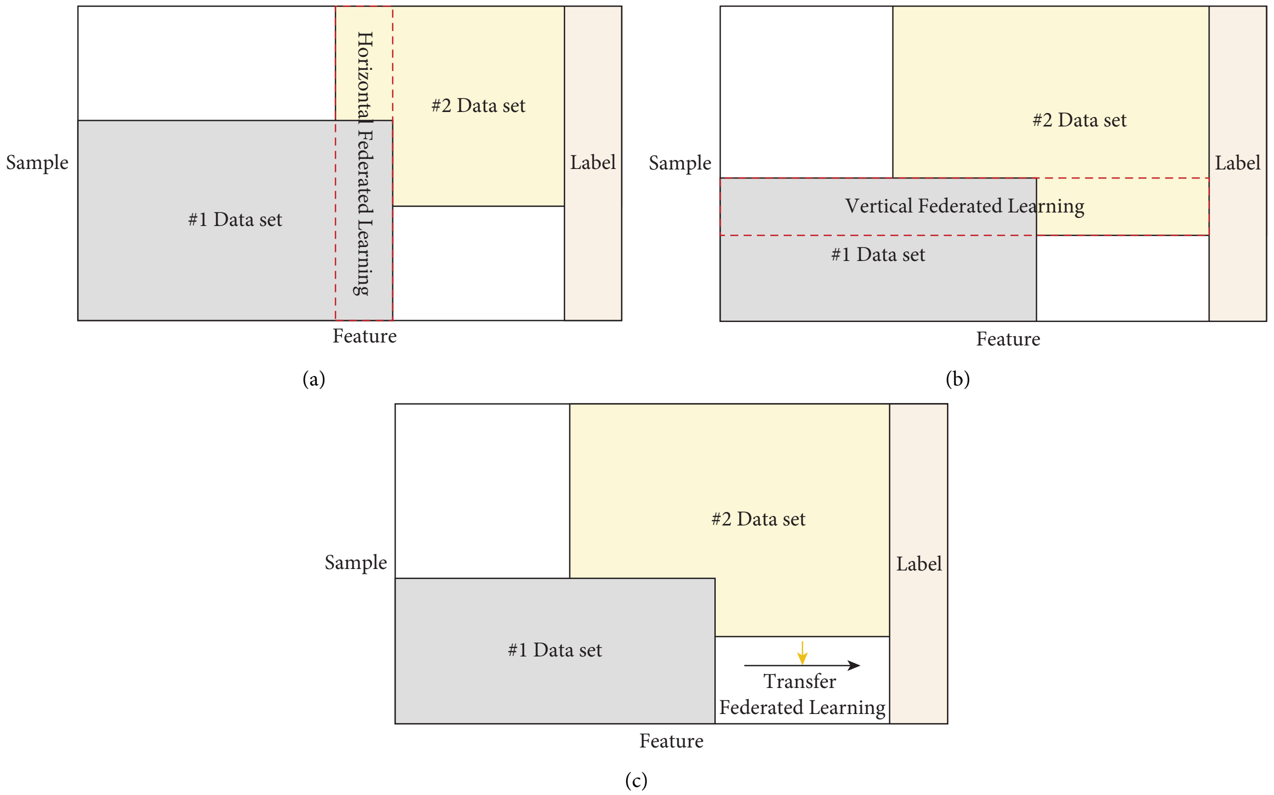

Figure 1: Different types of federated learning architectures. (a) Horizontal-federated learning. (b) Vertical-federated learning. (c) Transfer-federated learning.

different businesses but the same users, the same users indicate that their users will overlap greatly, and different businesses indicate that each organization has different user characteristics. When both parties want to jointly build a user consumption prediction model, due to the requirements of data privacy protection, neither party can directly obtain the other party's data. At this time, the two parties can use the vertical-federated learning technology to jointly build a model. Before performing longitudinal-federated learning, it is first necessary to find out the samples shared by the participants, that is, sample alignment. In longitudinal-federated learning, the participant's dataset can be viewed as being longitudinally sliced from an overall big data table. Vertical-federated learning expands the feature dimension of training samples. Vertical-federated learning is to combine different features possessed by participants without directly copying data to each other to enhance model accuracy. At present, vertical-federated learning has realized the modeling of many machine learning models such as neural network models, logistic regression models, and tree models. Therefore, vertical-federated learning can be summarized by

$$T_a \neq T_b, L_a \neq L_b, I_a = I_b, \forall D_a, D_b\, a \neq b. \tag{3}$$

3.2. Differential Privacy. Differential privacy is a strictly provable mathematical framework whose basic idea is to add carefully designed noise to the input or output of a function so that the modification of any single record in the dataset will not have a significant impact on the output. Thus, the attacker cannot infer the private information in datasets by analyzing the output results.

Differential privacy is defined as follows. Let $S\colon D \longrightarrow R$ be a random algorithm, D and D_1 are two datasets with at most one record different, $sO \in R$ is the output of algorithm S, and if algorithm S satisfies the following formula (4), it is said to satisfy (α, β) differential privacy.

$$P[S(D) = O] \leq e^{\alpha} P[S(D_1) = O] + \beta. \tag{4}$$

In formula (4), α is the differential privacy budget. The smaller the value is, the higher the degree of privacy protection is, but at the same time, the greater the accuracy loss of algorithm S is. β represents the probability that strict differential privacy is allowed to be violated, and the general value is small.

Sensitivity is defined as follows. For any query function $C\colon D \longrightarrow R^m$, D is the input dataset, and R^m is the m dimensional vector output by the function, then the sensitivity of function C is shown as

$$\Delta C = \max_{D,D_1} \left\| C(D) - C(D_1) \right\|_p. \tag{5}$$

In formula (5), D and D_1 are adjacent datasets that differ by at most one record, and $\|\cdot\|_p$ represents the L_p norm. Sensitivity reflects the maximum variation in the output of query function C on a pair of adjacent datasets. The smaller the sensitivity, the less noise needs to be added to the output to achieve differential privacy.

The definition of Gaussian mechanism is as follows. If the L_2 norm is used to calculate the sensitivity of function C, then differential privacy (α, β) can be achieved by adding Gaussian noise to the output of function C as

$$S(D) = C(D) + N\left[0, (\Delta C\sigma)^2 \overrightarrow{E}\right]. \tag{6}$$

In formula (6), Gaussian noise is a Gaussian distribution with a mean of 0 and a covariance of $(\Delta C\sigma)^2 \overrightarrow{E}$, and $\overrightarrow{E}$ is the identity matrix.

Differential privacy has the following two properties:

(1) *Postprocessing*. If the output of an algorithm satisfies differential privacy, any operation on this result will not cause additional privacy loss.

(2) *The principle of serialization combination*. The serialized combination of differential privacy algorithms still satisfies the differential privacy property.

3.3. Model Architecture. Aiming at the problems faced by the current indoor positioning methods, this paper proposes an indoor positioning privacy protection model based on federated learning in an MEC environment by comprehensively considering positioning performance, privacy protection, and resource overhead. The system architecture is shown in Figure 2.

The system architecture shown in Figure 3 is a 3-layer MEC framework, which divides the entire indoor positioning federated learning protocol into cloud server layer, edge server layer, and terminal device layer. Federated learning protocols with multiple participants are well supported. It is assumed that user group 1, user group 2, and user group 3 possess terminal devices and have collected a large amount of indoor positioning data, respectively. In order to be able to enjoy the indoor positioning service deployed on the edge server, they all voluntarily participate in the indoor-federated learning protocol. At the same time, they all try their best to prevent leaking their data to untrusted entities in the system (such as edge servers and cloud servers) during the entire federated learning process. The edge server performs aggregation and local submodel training after receiving the data that the terminal device has perturbed, and share the trained submodel parameters to the cloud server to obtain the optimal global positioning model. The cloud server receives the submodel parameters sent by the edge server, performs global model aggregation and collaborative update, and sends the updated model parameters to each edge server. The indoor positioning federated learning model is divided into two stages: offline training and online positioning. The specific description of entire system framework is as follows:

(1) *Terminal device*. It refers to a set of smart terminal devices (such as smartphones, tablet computers, smart monitoring equipment, and smart power terminals) owned by federated learning participants, with computing, storage, and communication capabilities. In the offline training phase, the terminal device can be used to obtain and store local datasets from multiple wireless sensor beacons in the indoor area and independently perform data preprocessing and noise addition on the collected datasets. The perturbed data are then sent to a nearby edge server. In the online positioning stage, edge devices send their measured real-time data, and it is perturbed with noise to the edge server to obtain positioning services.

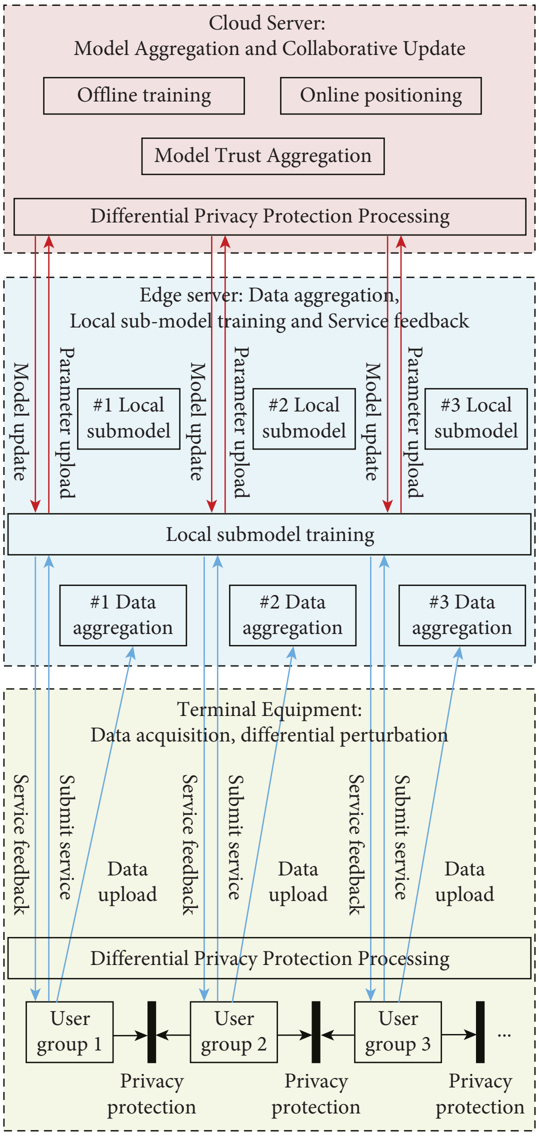

Figure 2: Indoor positioning privacy protection model based on federated learning in edge computing environment.

(2) *Edge servers*. This is the core entity of MEC architecture, usually implemented at user premises (such as parks, malls, and shopping centers) and may be deployed in fixed locations. They have more powerful storage and computing resources than terminal devices and act as computing units between cloud servers and terminal devices. Edge servers mainly perform trusted data aggregation, local submodel

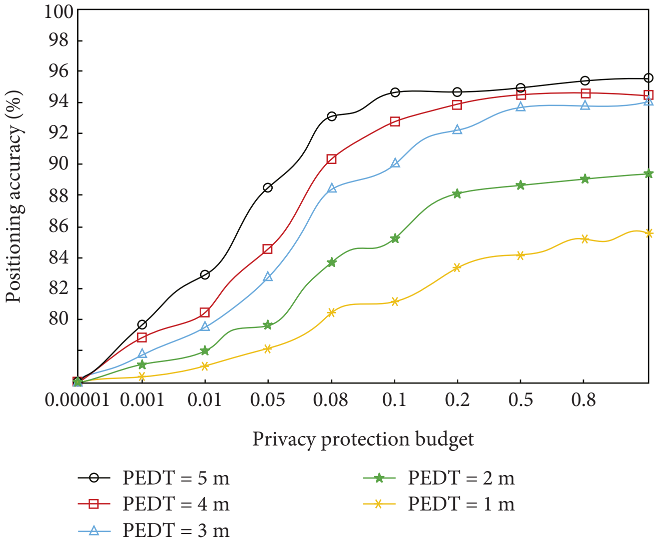

Figure 3: The positioning effect of the proposed algorithm under different privacy protection budgets.

training, and service feedback. In the offline training phase, the edge server first receives the perturbed data uploaded by the nearby terminal devices and aggregates the fingerprint data into data containing multiple user information. It uses these aggregated data to perform credible training of local positioning submodel, uploads the trained local submodel parameters to the cloud server, and repeats this iteration until the model converges. In the online positioning stage, the edge server uses the trained positioning model to provide users with high-credibility and high-precision indoor positioning services according to the real-time data submitted by users after privacy protection processing.

(3) *Cloud server*. As a data center, it has more powerful storage and computing power than edge servers. It receives the submodel parameters shared by each edge service, uses the federated average optimization algorithm to update the globally shared model parameters, and sends the updated model parameters to each edge server for the next round of iterative training until the optimal training model is obtained. In order to prevent untrusted cloud servers from inferring the private training data of each participating user through model inversion attacks or gradient reverse inference attacks, privacy protection processing is required when aggregating and updating global parameters. Here, differential privacy technology is used to add appropriate Laplace noise to the model parameters of each participant in the federated learning protocol and then the global parameters are aggregated and updated to achieve privacy protection. Among these entities, the end device is assumed to be trusted. It processes the collected data correctly and does not disclose it to other participants. Also, the edge servers and cloud servers are assumed to be honest and curious; that is, they can faithfully execute the federated learning protocol process and correctly compute and send real computation results. However, they are curious about the privacy contained in the data and do their best to analyze and mine the privacy of users. During the entire offline training process, the edge server only communicates with the cloud server. It cannot obtain any information about the rest of edge servers except for the jointly maintained global parameters, which guarantees the confidentiality of user data. In addition to privacy issues, the federated learning protocol in the MEC framework may also face the problem of resource constraints of end devices. Because executing complex deep learning models require huge computational overhead, resource-constrained terminal devices cannot afford the training process of complex deep learning models. Therefore, it is essential to design an effective positioning model that does not require too much computational overhead, does not violate the federated learning mechanism, and can protect the privacy of user data at the same time.

3.4. Algorithm Description.

Under the MEC architecture, users collect and send received signal strength (RSS) information to the cloud through intelligent edge devices to train the positioning model. Based on the proposed indoor positioning privacy protection model based on federated learning under MEC, a corresponding indoor positioning privacy protection method is proposed below. The method can update the user's local model and cloud model while providing (α, β) differential privacy protection and perform timely collaborative update in the form of device-cloud collaboration according to the location and demand changes of participating users.

The proposed algorithm mainly includes the following steps:

(1) The user collects RSS information data required for positioning on the terminal device and locally performs privacy protection processing on the private RSS information data that satisfies differential privacy. The processed data are then sent to nearby edge nodes.

(2) The edge nodes package and aggregate RSS information data sent by users, use the multiuser composite data of RSS information processed by the privacy protection mechanism to train the local submodel, and upload the trained model parameters to the cloud server.

(3) The cloud server performs differential private model aggregation on the local submodels from each edge node, confuses the contributions of each submodel to the global model, and obtains the cloud global model.

(4) Repeat steps (2) and (3) continuously, optimize and update the local submodel and cloud model, and finally realize the common benefits of each edge node.

On the terminal device, users can add controllable random noise $N(1/\delta_\mu)$ to their own private real RSS information data (x, y). Differential perturbation of RSS information data is performed before data sharing to ensure the privacy and security of RSS information data sent to edge nodes. RSS information data x_0 after differential perturbation can be expressed by

$$x_0 = x + N\left(\frac{\Delta C}{\delta_\mu}\right), \tag{7}$$

where $N(\cdot)$ represents the controllable random noise satisfying Laplace distribution, and the amount of added noise is controlled by the sensitivity ΔC. δ_μ denotes the privacy-preserving budget allocated to users for differential perturbation on end devices. The edge node packages the received RSS information data from different users into a local submodel training dataset (X_0, Y). where $X_0 = \{x_{01}, x_{02}, \ldots, x_{0n}\}$.

The cloud server receives local submodels uploaded from M different edge nodes and performs aggregation and update operations on these models that satisfy differential privacy protection. The update method of cloud model is shown as

$$G = \frac{1}{M}\left(\sum_{k=1}^{M} G_k + N\left(\frac{\Delta C}{\delta_\mu}\right)\right). \tag{8}$$

In formula (8), G represents the target parameter of cloud model. G_k denotes the k local submodel parameter, $k = 1, 2, 3, \ldots, M$. Based on this principle, it is continuously iterated to obtain the coevolution and update of cloud model and local submodel.

4. Experiments and Analysis

4.1. Simulation Environment and Datasets. In order to simulate the indoor positioning federated learning protocol in MEC environment, an indoor positioning model was built using TensorFlow, and two edge servers with the same amount of data were simulated. The socket protocol is used to realize the communication between the edge server and the parameter server, and the optimizer adopts AdaDelta. Instead of accumulating all past gradients, AdaDelta adjusts the learning rate based on the gradient update moving window without setting an initial learning rate. The number of iterations is 1000, and the batch size is 32. The hardware environment is as follows: Inter(R) Core(TM)i7-8750HCPU @2.20GHz, NVIDIA GeForce GTX1060 graphics card, and 24 GB RAM, 6 GB video memory.

The dataset employs real data collected using smartphones in a realistic indoor environment. The Wi-Fi access point AP and BLE beacon are preset in two indoor public areas to collect RSS information data, respectively.

The office area dataset is collected from a $12.5 \times 7.5\,\mathrm{m}^2$ office area. A total of 20 BLE beacons are deployed in the area, which can stably detect signals from 30 Wi-Fi APs. A total of 4232 samples were collected at the set 100 data collection points, and each sample contained two-dimensional position coordinates (x, y) and 50-dimensional RSS information features including 20-dimensional BLE features and 30-dimensional Wi-Fi features. The mall area dataset is collected from a shopping mall area of $32.5 \times 15.38\,\mathrm{m}^2$. The entire area is divided into 500 grid cells, as the size of each cell is $1\,\mathrm{m}^2$, and each grid cell is used as a data collection location point. A total of 22 BLE beacons are deployed near all collection locations, and signals from 35 Wi-Fi APs can be stably detected. A total of 9852 valid samples were collected, each containing 2D position coordinates (x, y) and 57D RSS information features including 22D BLE features and 35D Wi-Fi features.

4.2. Simulation Result Analysis. The (α, β) differential privacy guarantee provided by the algorithm results in a loss of localization accuracy. Based on the above datasets, first assume that the training sample labels are complete and the learning features are sufficient, and the positioning accuracy is only affected by the added controllable random noise. The preset error distance threshold is 1 meter to 5 meters, and the influence of different α on the positioning accuracy is compared under different error distance thresholds (PEDT), and the value range of α is from 0.00001 to 1. The positioning effect of the proposed algorithm under different privacy protection budgets is shown in Figure 3.

It can be seen from Figure 3 that the positioning accuracy increases with the increase of α. When α increases, the positioning accuracy changes significantly. This is because a smaller α means more noise is added and therefore a larger actual error distance. When $\alpha \geq 1$, the positioning accuracy almost reaches a steady state. This shows that by adding appropriate noise, the proposed algorithm can achieve good and stable localization accuracy while guaranteeing privacy. Under the same privacy budget, a larger PEDT corresponds to a higher localization accuracy. In the case of α, the positioning accuracy of proposed algorithm is about 95% when $\mathrm{PEDT} = 5\,\mathrm{m}$ and about 93% when $\mathrm{PEDT} = 3\,\mathrm{m}$. This is

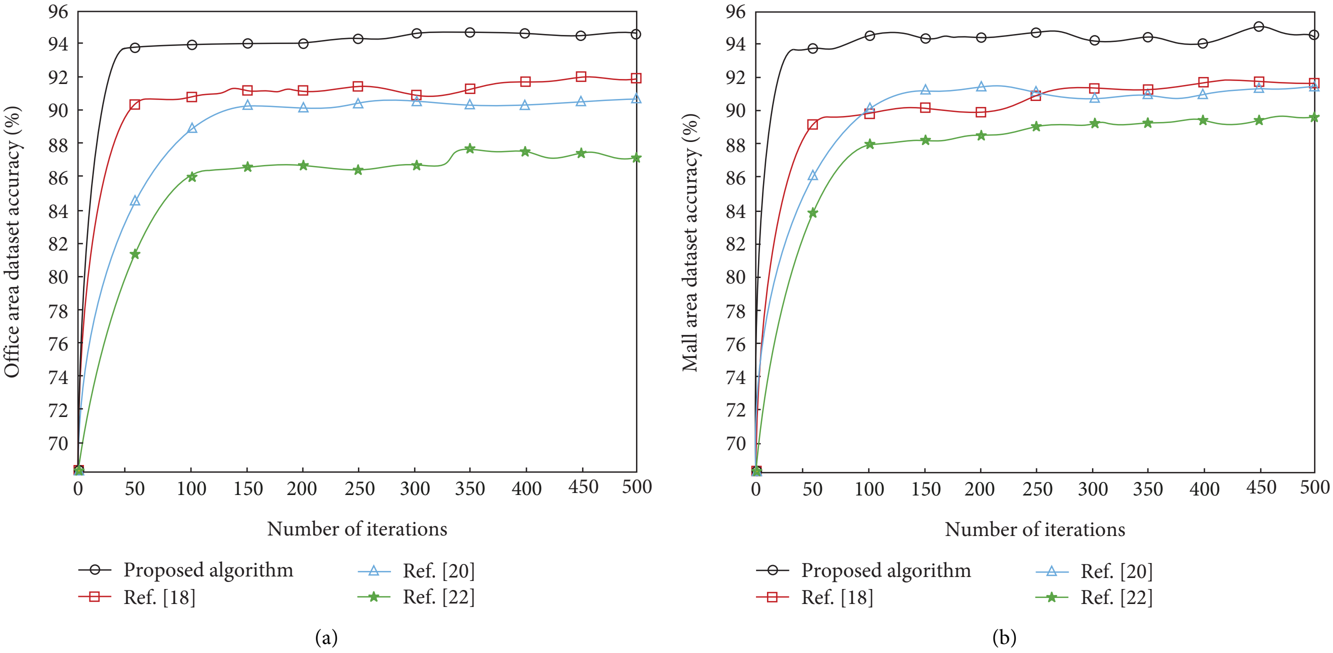

Figure 4: The positioning accuracy of different algorithms under the two datasets. (a) The training accuracy of different algorithms under the office area dataset. (b) The training accuracy of different algorithms under the mall area dataset.

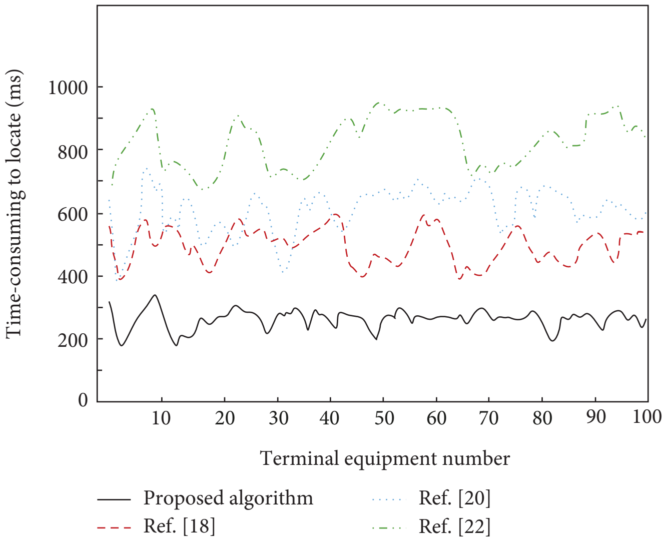

Figure 5: The time-consuming to locate different algorithms.

because in the case of adding the same noise, a larger PEDT means that the perturbed position satisfying the condition that the actual error distance is less than the preset error distance is more likely to be the correct result. A smaller value of α ensures higher privacy, which will also sacrifice more positioning accuracy and cause more time consumption. Therefore, in order to strike a balance between location privacy, localization accuracy, and time consumption, α is set to 0.1 in subsequent experiments.

4.3. Comparative Analysis. When the office area dataset and the mall area dataset are used, respectively, and the privacy protection budget is set to 0.1, the indoor positioning privacy protection method based on federated learning in edge computing environment is proposed in this paper using Refs. [18], [20], [22] for comparative analysis. The changes in the training accuracy of different algorithms are shown in Figure 4 below. The positioning time consumption of different algorithms is shown in Figure 5.

It can be seen from Figures 4 and 5 that with the increase in the number of training iterations, the positioning accuracy of all algorithms in the office area dataset and the mall area dataset gradually increases and finally tends to be stable, and the model becomes gradually stable convergently. Among them, the proposed algorithm has the fastest convergence speed and the highest positioning accuracy and starts to converge after 50 iterations and 40 iterations, respectively, under two different datasets, and their final positioning accuracy is about 94%, which is higher than the other three comparison algorithms, respectively.

In addition, for different user devices, the positioning completion time of proposed algorithm is the smallest, with an average of 250 ms, which is much lower than the positioning time of other three algorithms. This is because different types of federated learning for different situations can achieve the highest positioning accuracy and reduce time consumption through training without the data owner having to disclose its own data. Even when the edge server and cloud server are not trusted, it can still provide user training data privacy protection while resisting differential attacks, model inversion attacks, and gradient reverse inference attacks and obtain precise positioning accuracy. Different types of federated learning are used to process different private data, which improves the efficiency and reliability of private data processing. The introduction of differential privacy adds carefully designed noise to the input and output results so that the modification of any single record in the dataset will not have a significant impact on the output results. This makes it impossible for attackers to infer the privacy information in the dataset by analyzing the output results, thus achieving a better privacy protection effect.

5. Conclusion

Aiming at the problems of low positioning accuracy, long time-consumption, and difficulty to resist cross-attacks of traditional indoor positioning privacy protection methods, this paper proposes an indoor positioning privacy protection method based on federated learning in edge computing environment. The proposed method is verified by simulation experiments. The basic idea is as follows: (1) Firstly, the applicability of different types of federated learning mechanisms to datasets with different characteristics is analyzed and mathematically described, respectively. (2) Eliminate the possible damage to private data caused by federated learning by introducing differential privacy. (3) Build an indoor positioning privacy protection model and design the corresponding algorithm. Experimental results show that using different types of federated learning for private data with different characteristics can improve the efficiency and reliability of data processing. Besides, the introduction of differential privacy can greatly reduce the risk of leakage and destruction for user privacy data.

Comprehensive consideration of positioning performance, privacy protection, and resource overhead can greatly improve the indoor positioning accuracy of the algorithm. Due to the limitation of the dataset, there is no research on the classification of more modal data. In the future, we can carry out research on the simultaneous classification of more modal data. Future work will be devoted to using federated learning to train a unified neural network in the MEC scenario to achieve simultaneous classification of multiple modal data and the corresponding privacy protection schemes to serve diverse indoor positioning application requirements under the smart grid. Effectively improve the reliability and positioning accuracy of the indoor positioning algorithm and provide more accurate position information for the terminal equipment and power customer positioning demand services in the power production environment.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by 2022 Science and Technology Project of State Grid Hubei Electric Power Co., Ltd, "Research and Demonstrational Application of Key Technologies of Industrial Control Security Proactive Defense Architecture for Novel Power Systems" (No. 52153222001D).

References

[1] X. Zhu and Y. U. Jia, "Application of Beidou and UWB positioning technology in power system," *Shandong Electric Power*, vol. 48, no. 12, pp. 11–15, 2021.

[2] P. A. N. Guo-fu, O. Z-nan, and C. Bo-hao, "UWB/GNSS integrated positioning system design," *Navigation Positioning and Timing*, vol. 8, no. 4, pp. 129–134, 2021.

[3] C. Zhao, Y. L. Xi, and J. Wan, "Research on terminal location algorithm suitable for power production environment," *Automation & Instrumentation*, vol. 4, no. 12, pp. 12–15, 2017.

[4] J. Li, D. Xiong, and J. Cao, "Co-Location privacy protection method in mobile social networks," in *Journal of South China University of technology*, Natural Science Edition, Ed., vol. 47, no. 2, pp. 92–97, 2019.

[5] W. Lin, S. Zhang, and J. Liu, "Relay-assisted Offloading model for location privacy protection under edge computing," *Journal of Applied Sciences*, vol. 38, no. 5, pp. 724–741, 2020.

[6] Y. Qiu, Y. Liu, X. Li, and J. Chen, "A novel location privacy-preserving Approach based on Blockchain," *SENSORS*, vol. 20, no. 12, pp. 122–129, 2020.

[7] D. G. Feng, M. Zhang, and Y. T. Ye, "Research on differentially private Trajectory data Publishing," *Journal of Electronics and Information Technology*, vol. 42, no. 1, pp. 74–88, 2020.

[8] G. Du, L. Zhang, C. Ma, and G. Zhang, "Location privacy protection method based on attribute-based privacy information retrieval," *Journal of Harbin Engineering University*, vol. 42, no. 5, pp. 680–686, 2021.

[9] C. Wang, L. Zhang, and C. Zhang, "Privacy protection method for random transformation of adjacent sensitive areas," *Application Research of Computers*, vol. 37, no. 10, pp. 37–45, 2021.

[10] Z. Shen, Q. Zhang, and C. Zhang, "Location privacy attack based on deep learning," *Journal of Computer Research and Development*, vol. 59, no. 2, pp. 390–402, 2022.

[11] K. Q. Kou, Z. B. Liu, H. Ye, Z. Li, and W. Liu, “A location privacy protection algorithm based on differential privacy in sensor network,” *International Journal of Embedded Systems*, vol. 14, no. 5, pp. 432–442, 2021.

[12] Y. L. Zheng, J. Luo, and T. Zhong, “Service Recommendation Middleware based on location privacy protection in VANET,” *IEEE Access*, vol. 8, no. 23, pp. 12768–12783, 2020.

[13] J. D. Zhang and C. Y. Chow, “Enabling Probabilistic differential privacy protection for location Recommendations,” *IEEE TRANSACTIONS ON SERVICES COMPUTING*, vol. 14, no. 2, pp. 426–440, 2021.

[14] G. Song, G. Chu, and S. Wu, “Location privacy protection Approach based on Interval region,” *Computer Engineering and Application*, vol. 56, no. 8, pp. 66–73, 2020.

[15] Y. He, J. M. Zhang, L. S. Shuai, J. Luo, X. Yang, and Q. T. Sun, “A Personalized secure Publishing mechanism of the sensing location data in Crowdsensing location-based services,” *IEEE Sensors Journal*, vol. 21, no. 12, pp. 13628–13637, 2021.

[16] Y. Yang and R. Wang, “Location based service location privacy protection method based on location security in augmented reality,” *Journal of Computer Applications*, vol. 40, no. 5, pp. 1364–1368, 2020.

[17] C. Su, Y. M. Chen, and X. Z. Xie, “Location Recommendation with Privacy Protection,” in *Proceedings of the 2019 3rd International Conference on Intelligent Systems, Metaheuristics and Swarm Intelligence*, pp. 83–91, Male, MALDIVES, March 2020.

[18] Q. Y. Zhang, X. Zhang, M. Y. Wang, and X. Li, “DPLQ: location-based service privacy protection scheme based on differential privacy,” *IET Information Security*, vol. 15, no. 6, pp. 442–456, 2021.

[19] W. Wu, S. Fu, and Y. Luo, “Practical privacy protection scheme in WiFi fingerprint-based localization,” in *Proceedings of the 2020 7th IEEE International Conference on Data Science and Advanced Analytics*, pp. 699–708, Sydney, NSW, Australia, October 2020.

[20] H. Li, X. Ren, and J. Wang, “Continuous location privacy protection mechanism based on differential privacy,” *Journal on Communications*, vol. 42, no. 8, pp. 102–110, 2021.

[21] A. Zayets, C. Gentner, and E. Steinbach, “High-precision Multipath-based indoor localization scheme with user privacy protection for Dynamic NLoS environments,” *IEEE Access*, vol. 9, no. 12, pp. 116033–116049, 2021.

[22] X. Chen, T. Zhang, S. Shen, T. Zhu, and P. Xiong, “An optimized differential privacy scheme with reinforcement learning in VANET,” *Computers & Security*, vol. 110, no. 32, pp. 62–69, 2021.

[23] H. Jiang, J. Zeng, and K. Han, “Research on location privacy protection methods for mobile users in 5G Environment,” *Transactions of Beijing Institute of Technology*, vol. 41, no. 1, pp. 84–92, 2021.

[24] J. An, Z. X. Wang, X. He, X. Gui, J. Cheng, and R. Gui, “Know where You are: a practical privacy-preserving Semi-Supervised indoor positioning via edge-Crowdsensing,” *IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT*, vol. 18, no. 4, pp. 4875–4887, 2021.