EAI Endorsed Transactions
on Energy Web
Research Article EAI.EU

A Short term Electricity Load Forecasting for Community Residents Based on Federated Learning and Considering Privacy Protection

Bo Peng*, Jing Zuo, Xianfu Gong, Yaodong Li, Weijie Wu, Yixin Li

Guangdong Power Grid Co., Ltd. Power Grid Planning Research Center, Guangdong 510700, China

Abstract

INTRODUCTION: As the penetration rate of renewable energy increases and patterns of energy demand evolve, fluctuations on both the supply and demand sides of electricity are becoming more pronounced. Consequently, accurate forecasting of community residential electrical loads has become crucial.
OBJECTIVES: Although the widespread adoption of smart meters among residents provides abundant data for model training, strict challenges arise during the training process due to the need for privacy protection and data security.
METHODS: This paper proposes a privacy-preserving community residential short-term electric load forecasting method based on federated learning. Initially, the method applies shared random masking encryption to the sensitive data of community residents, ensuring data privacy while maintaining consistency with the original data after preprocessing. Subsequently, a private data aggregation scheme is established to perform dynamic clustering of the community's electrical load.
RESULTS: The clustered model then serves as the basis for establishing individual load forecasting models for each category of community residents to predict short-term electrical loads. Finally, an empirical analysis is performed using the electrical load data from 120 households across 6 communities in a city in Southern China.
CONCLUSION: The analysis demonstrates that the proposed method can achieve the prediction of community residential electrical loads without sharing residents' data, thus verifying the effectiveness of this approach.

Keywords: Neural Network, Privacy, Federated Learning, Aggregation, Short-term Load Forecasting

Received on 08 May 2024, accepted on 20 August 2024, published on 10 October 2024

doi: 10.4108/ew.6007

*Corresponding author. Email: bopengbp@outlook.com

1. Introduction

With the increasing scale of installations and total generation capacity of intermittent renewable energy sources such as wind and photovoltaics [1-3], short-term residential electricity load forecasting is becoming increasingly important for the reliability of modern power systems to address power security and system safety [4-6]. Short-term residential load forecasting covers periods from a few minutes to a week, and plays a significant role in many operational processes of the power system, such as the operation and dispatching of light storage charging and discharging equipment in residential areas [7].

Short-term residential electricity load forecasting relies on load data and configuration parameters [8,9]. Traditional residential load forecasts are mostly based on various regression analysis methods, including linear regression [10], autoregression [11], exponential smoothing [12], and time series [13] models. Data-driven forecasting methods rely heavily on the quantity and quality of available data [14]. With the widespread application of smart meters and the rapid development of machine learning methods, there has been a significant improvement in the quantity and quality of residential electricity load data. However, uploading large amounts of data to central servers not only increases the expenditure on communication resources but also poses risks such as the leakage of user privacy [15]. Short-term load

forecasting methods can use “centralized” or “decentralized” approaches for residential electric load prediction. The centralized approach involves transmitting smart meter data to a central server for concentrated training and forecasting. Its predictions are more accurate compared to the “decentralized” method, but it poses greater challenges regarding data privacy issues than the “decentralized” forecasting approach.

The “decentralized” approach refers to the use of distributed smart meter clients, each of which pre-trains and predicts based on their own historical data [16]. A widely applied method within this framework is based on federated learning. Federated learning enables decentralized clients to collaborate and update their own models without exchanging detailed data among themselves, thereby predicting their own futures [17]. In a typical federated learning framework, all clients cooperate in training under the coordination of a central server using their local data, to approximate a globally optimal model [18]. Since in federated learning the original data and computations are conducted on the local clients and the training process does not involve exchanging raw data between clients, it avoids the leakage of users’ private data. Furthermore, federated learning can combine multi-party computation and personalization techniques to further enhance privacy protection while also addressing performance degradation due to heterogeneous data across different users [19].

Machine learning models such as Extreme Gradient Boosting and Random Forests [20] are now widely applied in the field of time series prediction. At the same time, deep learning has been extensively utilized in the domain of electricity load forecasting due to the versatility of its models [21]. Deep learning methods have been employed to forecast electricity load intervals for regions, providing guidance for power dispatching [22]. Long Short-Term Memory (LSTM) networks have also been used for load time series prediction [23]. Based on LSTM, incorporating attention mechanisms can enhance the model’s memory capability and improve its interpretability [24]. These models have been comprehensively and extensively researched and experimented by scholars at home and abroad, and the results have verified the efficacy of machine learning methods in uncovering the complex non-linear relationships among factors influencing residential electricity load forecasting.

Load forecasting based on federated learning employs a variety of different neural network architectures. With the evolution of technology, the architectures of neural networks have become increasingly deep [25]. Deep neural networks include layers such as Fully Connected Layers (FCL) and Convolutional Neural Networks (CNN) [26]. LSTM has feedback connections that can understand the dependencies between sequence elements, making them suitable for time pattern recognition. The function of CNN layers is akin to the human retina, able to capture the spatial distribution of graphical patterns. In models that combine LSTM with CNN, within encoder-decoder or autoencoder architectures, the neural network takes a sequence vector as input and maps it to another sequence to minimize the impact of outliers and effectively avoid overfitting during training, with prediction performance indicators exceeding those of other machine learning models [27].

The standard federated learning approach establishes a single model for all clients. However, due to the inherent data heterogeneity of the decentralized clients, its performance is not as good as personalized federated learning. Personalized federated learning can learn from non-independent and identically distributed data, constructing customized models for each client. It employs data augmentation methods such as generative autoencoders to generate samples, reinforcing the statistical homogeneity from the local dataset’s perspective.

Despite significant progress in the industrial [28] and commercial sectors [29], the application of federated learning for residential electricity load forecasting has been scarce. Therefore, this paper explores the use of federated learning combined with cryptographic techniques and deep learning methods for residential electricity load forecasting. Specifically, to ensure the confidentiality of data in the proposed framework, a private data aggregation scheme was established. In addition, a dynamic clustering federated learning algorithm was proposed for different residential electricity load data, which iteratively groups users without leaking their privacy and shares inter-cluster information during the clustering process to achieve personalized forecasting for different types of residential electricity loads.

Addressing the issues mentioned above, this paper explores the potential application of federated learning in the field of power load forecasting and proposes a short-term electric load forecasting method based on federated learning with a focus on privacy protection. This method breaks away from the traditional system-level aggregate load forecasting approach, improving the performance of short-term residential electric load forecasting. Additionally, this study has developed an information-sharing framework based on federated learning to address issues of missing data and anomalies in residential electricity load data. Moreover, a privacy protection scheme for sensitive data aggregation has been established. Tailored to the heterogeneity of training data, a dynamic clustered federated learning prediction method has been implemented. Finally, using residential electricity load data from several communities in a city in Southern China as an example, the case study analysis shows that this method achieves electrical load prediction for residents while protecting data privacy.

2. Literature review

2.1. Characteristics of Federated Learning

Federated learning, as a distributed machine learning paradigm, is widely welcomed for its privacy-preserving characteristics [30]. A typical federated learning framework consists of a central server and multiple local clients [31]. In every communication round of federated learning, selected clients train the model with their local data independently,

and then broadcast their model updates to the central server. The central server then combines the updates from all clients to improve the global model on the central server. The training goal of federated learning is to provide local data with an approximation of a globally optimal model without disclosing user privacy. The main idea of federated learning is that multiple independent clients collaborate to train machine learning models without the need to exchange their detailed training data.

2.2. Forecasting Framework Construction

This research initially establishes a privacy-protected information sharing framework for electric load forecasting based on federated learning, the overall structure of which is depicted in Fig 1. Urban electric grid providers can fulfill the role of a central server, with urban residents participating in the information sharing framework, collaboratively enhancing forecasting performance without exchanging raw data. As illustrated in Fig 1, a private data aggregation scheme is first proposed to compute the statistical characteristics of the distributed data held by different residents while ensuring privacy during the model aggregation process; subsequently, based on the residential electric power load forecasting model, a dynamic clustering federated learning algorithm is proposed. This algorithm iteratively updates the clustering of participating resident users, enabling the sharing of classification information both within and among clusters. Meanwhile, to mitigate the impact of data heterogeneity, personalized forecasting models are developed for each category of residents.

The central server and all participating residents must adhere to the communication protocol established within the framework. During the client communication process, individual clients may attempt to analyze the exchanged data to obtain private information of other users [32]. Therefore, ensuring the privacy of users during the federated learning prediction process, and safeguarding that private data is not disclosed, becomes particularly crucial.

The input variables of the prediction model are as presented in Tab 1, which primarily include weather conditions, calendar time, residential housing features, and historical electric load characteristics. For numerical variables, the mean and standard deviation of the federated dataset are obtained using the proposed private data aggregation scheme, after which the data are normalized. Regarding categorical variables, those with periodic characteristics, such as the calendar, are encoded using sine and cosine functions to retain their cyclical nature. For other categorical features, such as the primary usage type of residences and time segments of the day, one-hot encoding is applied. To reduce the dimensionality of the one-hot encoded features, the time of day is divided into three main categories based on the characteristics of residential electricity use: “1” represents the time slot [9PM, 7AM), “2” represents [7AM, 5PM), and “3” signifies [5PM, 9PM).

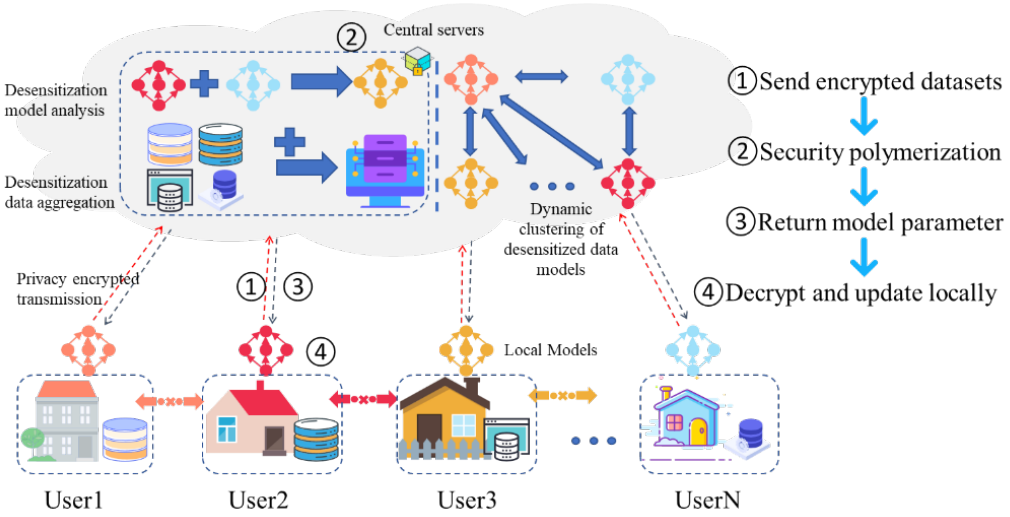


Figure 1 Federated-learning-based energy prediction architecture

Table 1 Input variables of the residential power prediction model.

Attribute	Variable	Type	Unit/Range
Environmental parameters	Temperature	Numerical	℃
	Humidity	Numerical	%
	Rainfall	Numerical	mm
	Atmospheric pressure	Numerical	hpa
	Wind speed	Numerical	m/s

	Wind direction	Numerical	Rad
Time attribute	Hour of the day	Categorical	0-23
	Day of the week	Categorical	1-7
	Weekend or Holiday	Boolean	True/False
	Day of the month	Categorical	1-31
Resident characteristics	Living area	Numerical	m^2
	type of Living	Categorical	1-4
Electric power demand	Power	Numerical	kW

Based on the features, the residential electricity load for the next hour is forecasted and the predicted results are then transformed using a specific function to reduce the variance of the forecasted outcomes. In terms of model input, given the clear periodic pattern observed in residential electricity load curves, the forecasting window is set to 24 hours to predict the next hour's electricity load using data from within this 24-hour window.

3. Federated Learning-based Residential Load Forecasting

Due to reasons such as data volume, traditional residential electricity load forecasting methods improve the performance of target residential load forecasts by utilizing models extracted from other resident categories. However, significant variations exist in load characteristics across different periods and residential areas. Moreover, machine learning models struggle to independently process and analyze data from different segments, leading to data heterogeneity issues. This heterogeneity severely hinders the construction of accurate residential load forecasting models. To address these challenges in a distributed framework, this paper proposes a dynamic clustering federated learning algorithm to cluster residents with similar load profiles.

3.1. Residential Load Forecasting Model Architecture

The LSTM network, as a type of recurrent neural network architecture, is a variant of the classic recurrent neural network and features the capability for long-term memory. The recurrent structure of the LSTM and its powerful memory cells excel in processing long-sequence time-series data [33]. Therefore, the LSTM is selected as the first component of the residential power load forecasting model to extract temporal features from input data. The internal structure of a cell is depicted in Fig 2, where σ_1 represents the forget gate; σ_2 and the hyperbolic tanh function form an input gate; σ_3 is the output gate; h_{t-1} denotes the output of the LSTM at the previous moment; x_t denotes the current input; h_t represents the current cell's output; and c_t and c_{t-1} respectively represent the current and previous moment's cell state.

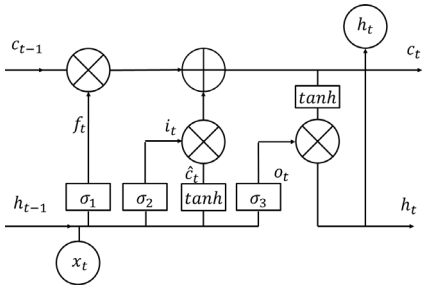


Figure 2 The Internal structure of LSTM

The sigmoid function and the hyperbolic tanh function are chosen as the gating units and hidden activation functions for the LSTM, respectively. To avoid overfitting, Dropout is employed as a regularization technique for each LSTM layer, omitting randomly selected LSTM units. The second block of the prediction model is a Multi-Layer Perceptron (MLP), which uses the output from the last time step of the LSTM as the input for the next module, establishing a mapping relationship between temporal features and residential electric load. Finally, the mean squared error loss function is used to measure forecasting errors and optimize network parameters. An L2 regularization term is added to the loss function to constrain the parameters to prevent overfitting. At time t, the LSTM cell processes are calculated as follows:

$$f_t = \sigma_1(W_f \bullet [h_{t-1}, x_t] + b_f) \quad (1)$$
$$i_t = \sigma_2(W_i \bullet [h_{t-1}, x_t] + b_i) \quad (2)$$
$$\hat{c}_t = tanh(W_c \bullet [h_{t-1}, x_t] + b_c) \quad (3)$$
$$o_t = \sigma_3(W_o \bullet [h_{t-1}, x_t] + b_o) \quad (4)$$
$$c_t = f_t c_{t-1} + i_t \hat{c}_t \quad (5)$$
$$h_t = o_t\, tanh(c_t) \quad (6)$$

In the aforementioned equations, $\boldsymbol{W_f}$、$\boldsymbol{W_i}$、$\boldsymbol{W_c}$ and $\boldsymbol{W_o}$ represent the matrix weights for the forget gate, input gate, candidate value vector, and output gate, respectively. Similarly, b_f、b_i、b_c and b_o denote the corresponding biases for each of these gates.

3.2. Dynamic Clustering FedL Algorithm

In a typical federated learning framework, the central server initially broadcasts the global prediction model parameters to all participating resident categories in each communication round. Subsequently, the resident clients use their local data to iterate and update their own model parameters. Finally, the

server aggregates all client models, iteratively updating to create the global prediction model.

Typical federated learning employs encrypted information sharing among resident categories, which ensures user privacy during the distributed training process and achieves the sharing of clustered information. However, this global aggregation method establishes a single model for all residents without considering the heterogeneity in residential loads. Hence, we propose the dynamic clustering federated learning algorithm, as shown in Fig 3. The process for intra-cluster information sharing among users and cluster optimization is as follows.

In the clustering process of residential electric load, traditional centralized clustering methods require the uploading of all residential load data to a central server, which poses a high risk of user privacy breaches and is unsuitable for scenarios requiring privacy protection. However, federated learning requires only the uploading of model parameters and encrypted data during the training phase, thereby enabling the clustering of residential electric loads without compromising user privacy. To group users with similar electric load characteristics into the same cluster, we use validation loss as the criterion for assessing the effectiveness of the clustering.

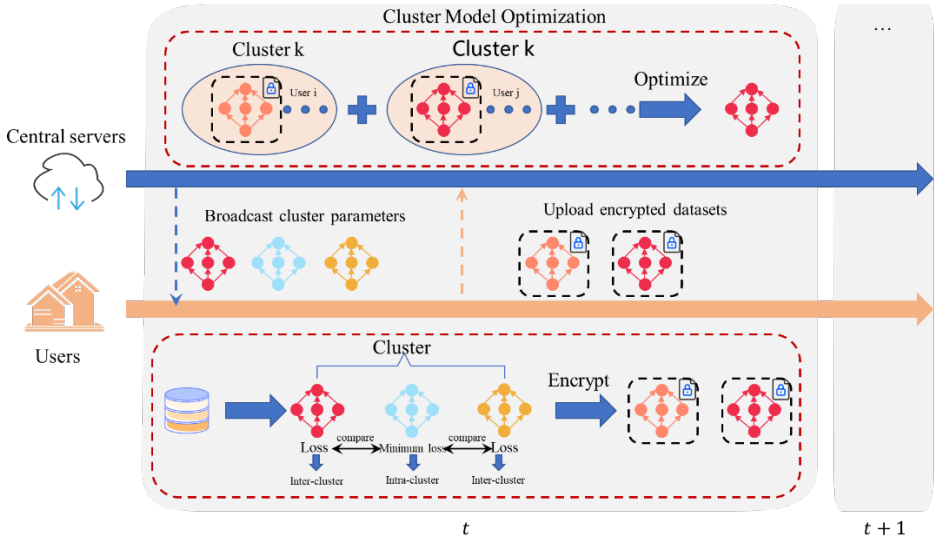


Figure 3 Federated learning Dynamical clustering algorithm.

3.2.1 Dynamic clustering method

The central server first generates initial model parameters θ_k^0 for K clusters, where $k=1,2...,K$. Then, under privacy constraints, clustering identification and cluster model optimization are performed for each global communication round. Each residential client can receive and updating the model k owned by the K clusters, and the residential clients only need to upload locally encrypted and trained data information.

During the global communication round in time t, the server initially assigns K model parameters to each participating resident. Subsequently, the i cluster residents perform local updates on the K cluster models $\theta_{i,k}^t$ they receive. That is, within the current period t, the local data set D_i is first divided into batches of size B. Then, the model parameters $\theta_{i,k}^t$ are updated utilizing the mean squared error loss function F and the Adam algorithm. The parameter update process for $\theta_{i,k}^{t,q}$ within the q period is expressed as follow:

$$m_q = \left(\beta_1 m_q + (1-\beta_1)\nabla\theta_{i,K}^{t,q}F_q\right)/(1-\beta_1) \quad (7)$$

$$v_q = \left(\beta_2 v_q + (1-\beta_2)\left(\nabla\theta_{i,K}^{t,q}F_q\right)^2\right)/(1-\beta_2) \quad (8)$$

$$\theta_{i,K}^{t,q+1} = \theta_{i,K}^{t,q} - a m_q/\left(\sqrt{v_q}+\varepsilon\right) \quad (9)$$

In the formula, F_q represents the loss value updated for period q, and $\nabla\theta_{i,k}^{t,q}$ is the gradient of the loss function with respect to the model parameters $\theta_{i,k}^{t,q}$. m_q and v_q are the bias-corrected estimates of the first-order moment and the second-order moment, respectively. a 、 β_1, and β_2 are the hyperparameters of the Adam optimization algorithm.

After E epochs of iterations, i cluster residents calculates the loss values for their cluster models. It is assumed that the list of loss values for i cluster residents during the global communication round at time t is denoted as L_i^t. i . The cluster residents is assigned to the cluster with the smallest validation loss from L_i^t which is the cluster of participants that share information within the cluster community, denoted as cluster k, represented as S_k.

In addition to intra-cluster information sharing, we also select some residents belonging to other clusters to optimize the predictive model of cluster k, thus implementing inter-cluster information sharing. The process of inter-cluster

information sharing can utilize more local data to participate in the optimization process of the cluster model. Suppose *i* cluster residents obtains the minimum loss for a cluster model in period *k'*, and the loss value of *i* cluster residents in period k's cluster is less than the product of the minimum loss value and the inter-cluster threshold γ, (γ≥1). Then resident *i* is added to the cluster of cluster *k*, namely the clustering S_k'. At the same time, the ratio of the minimum loss of the k-period cluster to the loss value is calculated as the decay factor $\eta_{i,k}^t \left(\frac{1}{\gamma} \le \eta_{i,k}^t \le 1\right)$. In the subsequent clustering optimization process, the decay factor is used as the weight coefficient for information sharing between different clusters.

3.2.2 Cluster model optimization

After clustering residents through the above process, we next utilize a federated learning algorithm to optimize each cluster model. Initially, the training model and encrypted data from *i* cluster residents are uploaded to the central server. Then, upon receiving information from the resident clients, the central server calculates the total number of samples for each cluster, that is $\sum_{j \in S_k \cup S_k'} |D_j|$ for cluster k, where *k*=1,2,…K. After receiving the broadcasted cluster sample sizes, the server adjusts the original model parameters $\theta^t_{i,k}$ to $\theta^t_{i,k,intra'}$ for the participation within cluster *k*, and adjusts the original model parameters $\theta^t_{i,k}$ to $\theta^t_{i,k',intra'}$ for the inter-cluster participation, with the modified expression as follows:

$$\theta_{i,k,intra}^t = \frac{|D_i|}{\sum_{j \in S_k \cup S_k'} |D_j|} \theta_{i,k}^t \qquad (10)$$

$$\theta_{i,k',intra}^t = \frac{|D_i|}{\sum_{j \in S_{k'} \cup S_{k'}'} |D_j|} \eta_{i,k'}^t \theta_{i,k'}^t \qquad (11)$$

The weights for the resident class i are given respectively by the proportion of the sample volume |Di| over the total number of samples participating in clusters k and k'. Concurrently, the decay factor $\eta^t_{i,k'}$ of the *i* cluster residents is incorporated into equation (11) to regulate the information sharing between clusters.

To prevent potential privacy breaches from the direct upload of model parameters, all residents engage in a random vector protocol to obtain the model parameters agreed upon for participation in clustering. After all users have uploaded their encrypted model parameters, the central server will perform model aggregation to optimize the federated cluster model. Specifically, within the global communication round of period t, the optimized model for cluster k is:

$$\theta_k^{t+1} = \sum_{i \in S_k \cup S_k'} \left(\theta_{i,k,intra}^t + \omega_{i,k}^t\right) \qquad (12)$$

Wherein, $\omega^t_{i,k'}$ represents the shared model parameters for the *i* cluster residents, participating in the information-sharing model for cluster k in encrypted form. Expression (12) ensures that the output is consistent with the results of direct aggregation models while also mitigating the potential risk of privacy leakage associated with the original models.

3.3 Experiment setup

Firstly, the model proposed in this study is used to predict residential electricity load, while the Adam algorithm is employed to optimize the model parameters during training. Additionally, grid search is utilized to adjust the hyperparameters of the optimized model, thereby enhancing its predictive performance. The detailed grid search range settings are presented in Tab 2.

Table 2 Search ranges of hyperparameters for Power prediction approach

Aspects	Property	Values
Model architecture	LSTM layers	2,3
	LSTM units in each layer	32,64,128,256
	MLP layers	2,3
	MLP units in each layer	32,64,128,256
Optimizer and regularization	Learning rate	0.001,0.001, 0.1
	Dropout probability	0.5,0.6
	L2 regularization	0.0001,0.0005
Dynamical clustering federated learning	Local epochs	5,6,7,10,15
	Local batch	32,64
	Cluster number	3,4,5
	Inter-cluster threshold	1.0,1.1,1.2

3.4 Data preprocessing

To enhance the usable quality of the dataset, it is imperative to preprocess the raw data. The main aspects of data preprocessing comprise data cleaning, data transformation, and data standardization [34]. This preprocessing was conducted in three principal stages. Initially, we reduced the frequency of time-series data to alleviate the computational demand of the simulation. Secondly, outliers and missing values were removed from the dataset. Finally, min-max scaling was used to normalize all variables, ensuring the operability of residential electric load forecasting models.

Direct usage of statistical characteristics of resident users would undoubtedly lead to privacy breaches. However,

within federated learning, the models and methods for aggregation or gradient aggregation could, by reconstructing the training models or data uploaded by users, restore original user information, thereby also leading to privacy breaches. Consequently, there is a need for a method that can normalize the model parameters and training results uploaded by clients without compromising privacy while preserving the integrity of the original information.

Suppose there are N households of residents, and their local training datasets are D_1, D_2, …D_N. Z-score normalization is employed to process numerical variables and to calculate the mean $\mu = [\mu_1, \mu_2, \ldots, \mu_c]^T$ and standard deviation $\sigma = [\sigma_1, \sigma_2, \ldots, \sigma_c]^T$ with respect to the numerical features on the entire training dataset C where $D=D_1 \cup D_1 \cup \ldots \cup D_N$,. Subsequently, the calculated mean and standard deviation are used to normalize the test set, to avoid potential future information leakage. The normalization method for variable x_i is demonstrated as follows:

$$\hat{x}_i = \frac{x_i - \mu_i}{\sigma_i} \quad (13)$$

$$\mu_i = \frac{1}{M}\sum_{j=1}^{N}\sum_{x_i \in D_j} x_i \quad (14)$$

$$\sigma_i = \sqrt{\frac{1}{M-1}\left(\sum_{j=1}^{N}\sum_{x_i \in D_j}(x_i)^2 - M\mu_i^2\right)} \quad (15)$$

Where $\hat{x}_i$ is the normalized value, $M = \sum_{j=1}^{N}|D_j|$ is the sum of all samples in the local training datasets, and $|D_j|$ is the size of the samples in the local training dataset D_j The sample size for each residential class is not sensitive information, hence the central server is able to obtain M by directly aggregating the sizes of the local training dataset samples.

$$X^i = \left[\sum_{x_1 \in D_1} x_1, \ldots, \sum_{xc \in D_1} x_c, \sum_{x_1 \in D_1}(x_1)^2, \ldots, \sum_{xc \in D_1}(x_c)^2\right] \quad (16)$$

However, the uploaded vector $\boldsymbol{X^i}$ can reflect the load pattern of the i cluster residents, but this may also lead to the leakage of user privacy. Therefore, this study establishes a private data aggregation scheme based on pairwise masking and the Elliptic Curve Diffie–Hellman (ECDH) protocol. This scheme can protect user privacy while ensuring that the mean and standard deviation of the encrypted dataset are consistent with those of the original data. The overall architecture is illustrated in Fig 4.

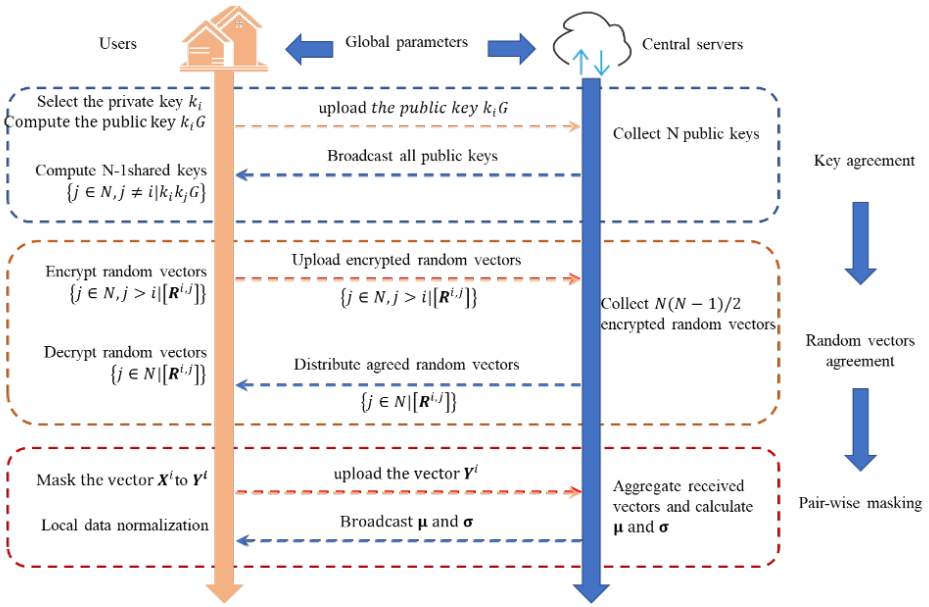


Figure 4 Private data aggregation algorithm.

3.4.1 ECDH Key Agreement Protocol

The ECDH key agreement protocol allows classes of residents to generate a shared key over an unsecured communication channel. As depicted in Fig 4, the main parameters of ECDH include the elliptic curve, base point G, prime p, order n, and cofactor h of the subgroup, all of which are initially broadcasted to all classes of residents by the server. Based on the same domain parameters, each class of resident i selects an integer $k_i (k_i \in [2, n-1])$ as the private key for computation, and then uploads the public key k_iG to the central server. The central server, acting as an intermediary communication node, collects all public keys and distributes them to all residents. Each resident class i computes N-1 shared keys ($j \in N$, $j \neq i | k_i k_j G$), thereby reaching a consensus on pairwise random vectors. The security of the ECDH key agreement protocol is ensured by the elliptic curve discrete logarithm problem. Although the central server possesses the base point G and the public keys k_iG, it cannot solve for the private keys k_i, thus avoiding the leakage of user privacy.

3.4.2 Random vectors agreement

Each class of residents i selects two dimensional random vectors denoted as $R^{i,j}(j \in N, j > i)$, which are encrypted into $[R^{i,j}]$ based on the shared key $k_i k_j G(j \in N, j > i)$. Once the server receives all pairs of encrypted random vectors, resident users of class i will decrypt the corresponding encrypted random vectors to obtain $R^{i,j}(j \in N)$, effectively substituting the privacy-sensitive original data $\boldsymbol{X^i}$.

3.4.3 Pair-wise masking

The method of pairwise masking involves hiding the actual values of the original data $\boldsymbol{X^i}$ by adding random vectors, thereby ensuring the accuracy of the calculations for the mean and the standard deviation of the entire dataset. Based on the shared random vectors $R^{i,j}(j \in N)$, $\boldsymbol{X^i}$ can be transformed into a masked form $\boldsymbol{Y^i}$, and the transformation model is shown as Equation (17):

$$Y^i = X^i + \sum_{j \in N, j>i} R^{i,j} - \sum_{j \in N, j<i} R^{i,j} \quad (17)$$

Subsequently, the i cluster residents upload the masked vector $\boldsymbol{Y^i}$ to the server. The server aggregates the complete set $\boldsymbol{Y^i}(i \in N)$ for the entire dataset D and can obtain the mean μ of the entire dataset using Equation (14) and the standard deviation σ using Equation (15). The server then broadcasts μ and σ to all resident classes for the normalization of user data. The central server validates the correctness of $\boldsymbol{X^i}$ through aggregation operations, and the verification equation is as follows:

$$X = \sum_{i=1}^{N} Y^i = \sum_{i=1}^{N} X^i + \sum_{i=1}^{N} \left(\sum_{j \in N, j>i} Y^i - \sum_{j \in N, j<i} R^{i,j} \right) = \sum_{i=1}^{N} X^i \quad (18)$$

As demonstrated above, the private data aggregation scheme does not require communication between clients or an additional trusted third party, thereby avoiding potential risks of privacy leakage.

4. Case Studies

4.1. Experiment and dataset description

This chapter validates the effectiveness of the proposed scheme through numerical examples. The experimental environment is described as follows: the CPU is an AMD EPYC 9554 64-Core Processor with a frequency of 3.10 GHz, the GPU is an NVIDIA RTX 4090 with 24GB of video memory, the system RAM is 64GB, the operating system used is Windows 10. The Pytorch version implemented is 2.1.2, CUDA version is 12.1, and Python is at version 3.8.18. Federated learning is executed based on the FedML library. The hyperparameters for the load forecasting model are as indicated in Table 2. The simulation dataset comprises data from 120 households within six communities in a southern city of China, with the data precision being one data point every half hour. The dataset also includes local weather conditions for the residents of these communities. The data is split according to an 80:20 ratio to create training and validation sets, respectively.

4.2. Evaluation metrics

To assess the forecasting performance of the model, it is necessary to employ relevant evaluation metrics to compare the performance of the proposed model. Common metrics include the Coefficient of Variation (CV), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Root Mean Square Error (RMSE) to evaluate the performance of the residential electrical load forecasting model. The expressions for these four metrics are as follows:

$$CV = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)} \Big/ \frac{\sum_{i=1}^{n} y_i}{n} \quad (19)$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i| \quad (20)$$

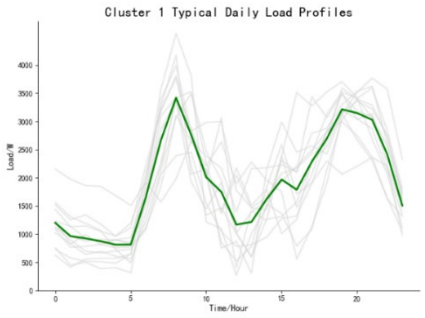
$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{n}\right| \quad (21)$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \quad (22)$$

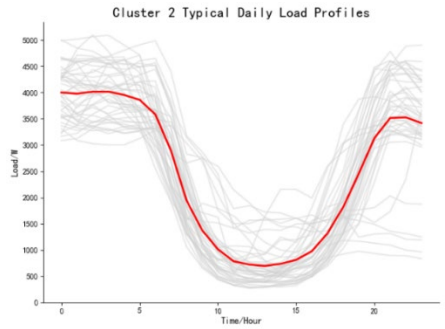
In the formulas, n represents the number of samples in the test dataset, which is also the number of participating users. $\hat{y}_i$ and y_i are the predicted and actual residential electrical loads, respectively. For MAE, MAPE, and RMSE, the smaller the results, the better the forecasting performance.

4.3. Experiment Results and Analysis

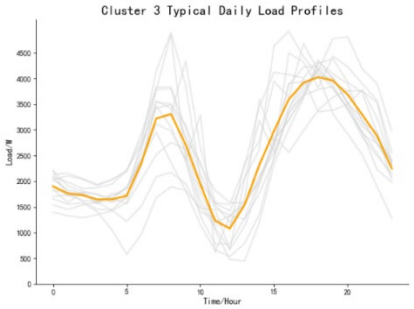
In this numerical example, clustering is first performed based on the electrical load characteristics of 120 households across six communities. According to the characteristics of the resident's load, they can be divided into four types, as illustrated in Fig5. The parameters of the forecasting model for each type of user are optimized according to Tab 2. Throughout the entire simulation process, there are a total of ten interaction rounds between the central server of the urban electric grid and the community resident users. Here, an interaction round refers to the number of times the central server of the urban electric grid updates the global weight parameters with the participating residents.



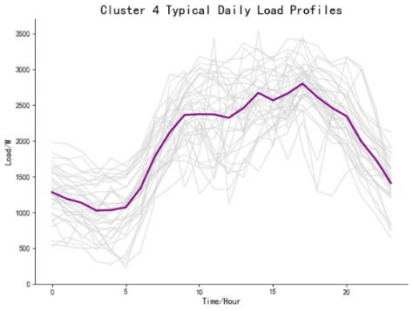
(a) Cluster 1 Typical Daily Load Profiles



(b) Cluster 2 Typical Daily Load Profiles



(c) Cluster 3 Typical Daily Load Profiles

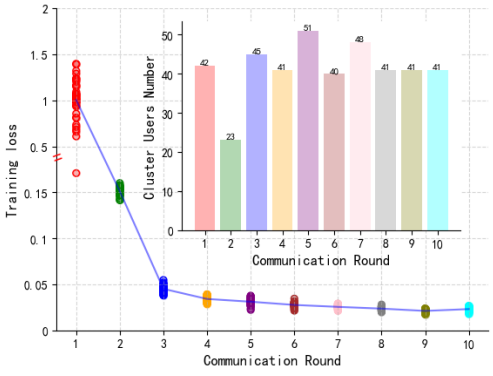


(d) Cluster 4 Typical Daily Load Profiles

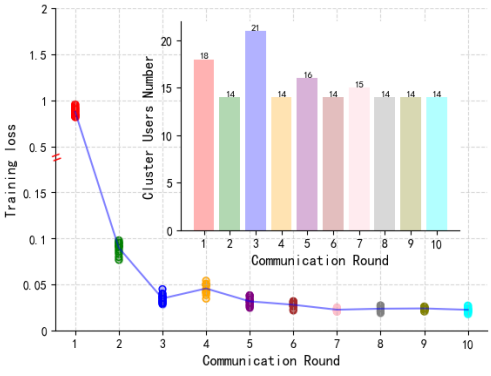
Figure 5 Cluster typical load profiles of users.

4.3.1 Convergence Process

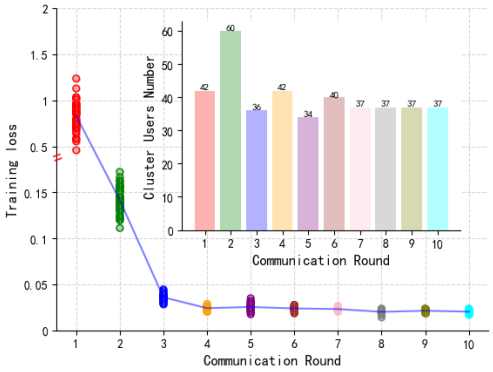
This study initially employs a dynamic clustering federated learning algorithm, performing iterative clustering based on residents' characteristics and engaging in intra-cluster and inter-cluster anonymous information sharing and interactive training. Fig 6 vividly illustrates the convergence process across various clusters. In the Fig6, each point represents the training loss value of users in each communication round, while the curve denotes the average training loss of all residents belonging to that cluster category. The histogram delineates the change in the number of resident users within the corresponding resident class during the dynamic grouping process. As depicted in Fig 6, the average training loss for each cluster shows a generally monotonous decreasing trend and drops rapidly in the initial periods, indicating the proposed dynamic clustering federated learning algorithm converges swiftly. Furthermore, as the iterative clustering process progresses in Fig 6, the number of constructed clusters stabilizes gradually.



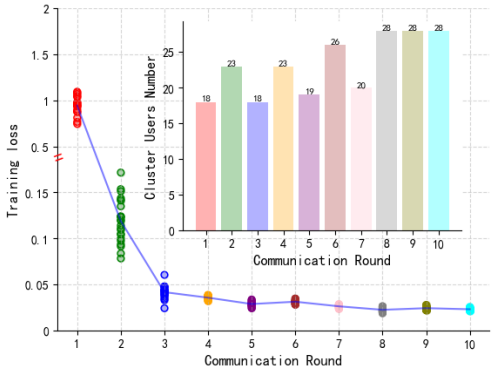
(a) Residential cluster 1



(b) Residential cluster 2



(c) Residential cluster 3



(d) Residential cluster 4

Figure 6 The convergence of the dynamic clustering federated learning algorithm.

4.3.2 Performance Comparison

To verify the effectiveness of the proposed forecasting approach, this paper has conducted a comprehensive comparison of the method with traditional local forecasting and centralized forecasting approaches. Local forecasting refers to the scenario where residents' local clients train models and make predictions based on their own data without the need to communicate with a central server. The local forecasting approach employs a two-layer LSTM, each with 32 neurons, to serve as the prediction model to avoid the phenomenon of overfitting. Conversely, centralized forecasting involves aggregating all residential electricity load data to establish a single prediction model for all residents. The method proposed in this paper is based on federated learning and takes privacy protection into account for electricity load forecasting. Table 3 presents the validation results and average values of all evaluation metrics (CV/MAE/MAPE/RMSE) for the proposed method compared to the other two methods when predicting electricity load for all residents. The local prediction method provides good protection of residents' privacy but has relatively high error metrics. Although the central prediction method has certain performance advantages over the local prediction method, it leads to user privacy leakage. The federated learning method proposed in this study not only significantly outperforms the traditional local prediction method in terms of prediction performance but also effectively protects user privacy.

Table 3 Performance metrics of three different approaches for Residents power prediction.

Method	CV	MAE	MAPE	RMSE	Privacy guarantee
Proposed method	0.0661	7.3164	0.0513	9.1313	True
Local prediction	0.1314	13.2181	0.0923	16.1932	True
Centralized prediction	0.0978	12.1123	0.0821	14.8918	False

5. Conclusions

To enhance the performance of residential electric load forecasting while ensuring user data privacy, this paper proposes a community residential short-term electric load forecasting method based on federated learning with consideration for privacy protection. First, perform dynamic clustering analysis of community residential load characteristics, then for each category of users, select weather, time, and resident characteristic factors based on the LSTM model to establish a load forecasting model. Finally, use distributed local prediction algorithms, central prediction algorithms, and this method to forecast and comparatively analyze the community residential electric load. The results of the numerical example demonstrate that federated learning can be combined with privacy protection technologies to forecast the short-term electric load of community residents.

Certainly, this research has certain limitations. In the process of clustering analysis, it only classified users into four categories based on their load characteristics. Future research can perform clustering analysis by combining more characteristics, which would allow the forecasting model to learn more user features, thereby improving the accuracy of predictions.

Acknowledgements

This work was supported by the Science and Technology Program of Science and Technology Project of China Southern Power Grid Co., Ltd. (030000KC23040062(GDKJXM20230367)).

References

[1] ZHAO Hanting, ZHANG Yao, HUO Wei, et al. Collaborative Forecasting Method for Short-term Wind Power Based on Vertical Federated Learning. Automation of Electric Power Systems. 2023, 47(16): 44-53.

[2] ZHENG Jie, NIU Zhewen, HAN Xiaojie, et al. Facing Data Privacy Protection in Distributed Short-Term Wind Power Forecasting for Multiple Wind Farms. Journal of Taiyuan University of Technology.2024, 55(01): 102-110.

[3] GAO Yi, ZHOU Yu, ZHANG Anlong, et al. The PV output and load power prediction based on personalized federated learning under the photovoltaic system in the whole county. Power System Technology. 2023, 47(11): 4629-4638.

[4] ZHU Congyang, ZHANG Ge, JIA Yujing, et al. Based on the Long Short-Term Memory Network Model, Federated Learning for Residential Load Forecasting. Modern Electric Power. 2023.

[5] HUA Yuanpeng, WANG Yuanyuan, HAN Ding, et al. Mid-and Long-term Charging Load Forecasting for Electric Vehicles in Residential Areas Considering Orderly Charging. Proceedings of the CSU-EPSA. 2022, 34(6): 142-150.

[6] DONG Tao, YONG Jing, ZHAO Jin, et al. The comprehensive management system of power load for residential areas with PV, energy storage, and EVs. Journal of Chongqing University. 2021, 44(08): 45-58.

[7] LIU Xiaofeng, GAO Bingtuan, LUO Jing, et al. A model of residential load stratification scheduling based on non-cooperative game theory. Automation of Electric Power Systems. 2017, 41(14): 54-60.

[8] KONG Xiangyu, MA Yuying, AI Qian, et al. Review on Electricity Consumption Characteristic Modeling and Load Forecasting for Diverse Users in New Power System. Automation of Electric Power Systems. 2023, 47(13): 2-17.

[9] LU Jixiang, ZHANG Qipei, YANG Zhihong, et al. Short-term load forecasting method based on CNN-LSTM hybrid neural network model. Power System Automation 2019, 43(08): 131-137.

[10] WANG Dewen, SUN Zhiwei. Power user-side big data analysis and parallel load forecasting. Proceedings of the Chinese Society for Electrical Engineering. 2015, 35(03): 527-537.

[11] SUN Qingkai, WANG Xiaojun, ZHANG Yizhi, et al. Based on LSTM and multi-task learning, multi-dimensional load forecasting for integrated energy systems. Power System Automation 2021, 45(05): 63-70.

[12] GE Xiaolin, SHI Liang, LIU Ya, et al. Consideration of demand response uncertainty for EVs load Sigmoid cloud model prediction. Transactions of China Electrotechnical Society. 2020, 40(21): 6913-6925.

[13] LIU Xin, LIU Donglan, FU Ting, et al. An algorithm for time series prediction based on federated learning. Journal of Shandong University. 2023.

[14] CAO Zhaojing. The research on source-load probability prediction and its application in the new energy power system driven by data-model fusion. Zhejiang University.2022.

[15] WANG Teng, HUO Zheng, HUANG Yaxin, et al. Review on privacy-preserving technologies in federated learning. Journal of Computer Applications. 2023, 43(2): 437-449.

[16] Zhang L, Wen J, Li Y, et al. A review of machine learning in building load prediction. Applied Energy. 2021, 285: 116452.

[17] Mothukuri V, Parizi R M, Pouriyeh S, et al. A survey on security and privacy of federated learning. Future Generation Computer Systems. 2021, 115: 619-640.

[18] LIU Xiaoqian, XU Fei, MA Zhuo, et al. Research on Privacy Protection Technology in Federated Learning. Journal of Information Securyity Research. 2024, 10(03): 194-201.

[19] Li L, Fan Y, Tse M, et al. A review of applications in federated learning. Computers & Industrial Engineering. 2020, 149: 106854.

[20] Wang Z, Wang Y, Zeng R, et al. Random Forest based hourly building energy prediction. Energy and Buildings. 2018, 171: 11-25.

[21] ZHU Youcheng, WANG Jinrong, XU Jian. Medium and Long Term Wind Power Generation Forecasting Method Based on Deep Learning. Guangdong Electric Power. 2021, 34(6): 72-78.

[22] YU Dengwu, LIU Min, PU Fannuo,et al. Power Load Interval Prediction Method Based on Deep Learning Quantile Regression. Guangdong Electric Power. 2022, 35(09): 1-8.

[23] Somu N, M R G R, Ramamritham K. A hybrid model for building energy consumption forecasting using long short term memory networks[J]. Applied Energy. 2020, 261: 114131.

[24] Li A, Xiao F, Zhang C, et al. Attention-based interpretable neural network for building cooling load prediction. Applied Energy. 2021, 299: 117238.

[25] Kim T, Cho S. Predicting residential energy consumption using CNN-LSTM neural networks. Energy. 2019, 182: 72-81.

[26] Le T, Vo M T, Vo B, et al. Improving Electric Energy Consumption Prediction Using CNN and Bi-LSTM. 2019: 9.

[27] QIN Feixiang, ZHU Gelan. Falut Loacation MEthod for Distribution Network Based on LSTM-CNN Machine Learning. Guangdong Electric Power. 2021, 34(11): 27-34.

[28] WANG Beibei, ZHU Jing, WANG Jiale, et al. Federated-learning Based Industry Load Forecasting Framework Under Privacy Protection of Meter Data. Automation of Electric Power Systems. 2023, 47(13): 86-93.

[29] CHEN Haoyu, LI Yidong, ZHANG Honglei et al. A research review on fairness in federated learning. Acta Electronica Sinica. 2023, 51(10): 2985-3010.

[30] WANG Jianzong, KONG Lingwei, HUANG Zhangcheng, et al. Research review of federated learning algorithms. Big Data Research. 2020, 6(6): 64-82.

[31] C Jiayi, S Chenyu, Z Xintong, et al. Local Protection of Power Data Prediction Model Based on Federated Learning and Homomorphic Encryption. Journal of Information Securyity Research. 2023, 9(3): 228-234.

[32] Eibl G, Engel D. Differential privacy for real smart metering data. Computer Science - Research and Development. 2017, 32(1): 173-182.

[33] CHEN Zhenyu, LIU Jinbo, LI Chen, et al. Based on the combination model of LSTM and XGBoost for ultra-short-term electricity load forecasting. Power System Technology 2020, 44(02): 614-620.

[34] Feature Engineering for Machine Learning. Beijing: Posts and Telecommunications Press, 2019: 156.