BIG DATA MINING AND ANALYTICS
ISSN 2096-0654 03/10 pp421–432
Volume 6, Number 4, December 2023
DOI: 10.26599/BDMA.2022.9020043

Personalized Federated Learning for Heterogeneous Residential Load Forecasting

Xiaodong Qu, Chengcheng Guan, Gang Xie*, Zhiyi Tian, Keshav Sood, Chaoli Sun, and Lei Cui

Abstract: Accurate load forecasting is critical for electricity production, transmission, and maintenance. Deep learning (DL) model has replaced other classical models as the most popular prediction models. However, the deep prediction model requires users to provide a large amount of private electricity consumption data, which has potential privacy risks. Edge nodes can federally train a global model through aggregation using federated learning (FL). As a novel distributed machine learning (ML) technique, it only exchanges model parameters without sharing raw data. However, existing forecasting methods based on FL still face challenges from data heterogeneity and privacy disclosure. Accordingly, we propose a user-level load forecasting system based on personalized federated learning (PFL) to address these issues. The obtained personalized model outperforms the global model on local data. Further, we introduce a novel differential privacy (DP) algorithm in the proposed system to provide an additional privacy guarantee. Based on the principle of generative adversarial network (GAN), the algorithm achieves the balance between privacy and prediction accuracy throughout the game. We perform simulation experiments on the real-world dataset and the experimental results show that the proposed system can comply with the requirement for accuracy and privacy in real load forecasting scenarios.

Key words: load forecasting; personalized federated learning; differential privacy

1 Introduction

Electric power cannot be stored as a special commodity. For this reason, it is crucial to the power system's stability that ensures the balance of power supply and demand. With timely and accurate load forecasting results, power producers can effectively manage the production, transmission, and distribution of electricity.

Power load forecasting has been a concern by researchers because of its importance in the power grid. At the beginning of the research, both time series analysis and regression analysis methods[1] were used for power load forecasting. However, these classical methods have limited ability to deal with problems and are inadequate in nonlinear data. In the 21st century, machine learning (ML) and deep learning (DL) have been employed in many fields of intelligent computing with good performance[2]. In general, ML-based prediction methods require a great deal of experience to cultivate prediction models. Moreover, ML also faces many challenges in feature selection, time series issues, and sample complexity[3]. The smart grid (SG)[4] is an

• Xiaodong Qu, Chengcheng Guan, Gang Xie, and Lei Cui are with the Shanxi Key Laboratory of Advanced Control and Equipment Intelligence, Taiyuan University of Science and Technology, Taiyuan 030024, China. E-mail: {S20201503013, S202115110206}@stu.tyust.edu.cn; {xiegang, lei.cui}@tyust.edu.cn.

• Zhiyi Tian is with the Faculty of Engineering and Information Technology, University of Technology Sydney, Ultimo 2007, Australia. E-mail: zhiyi.tian@student.uts.edu.au.

• Keshav Sood is with the Centre for Cyber Security Research and Innovation, Deakin University, Melbourne 3125, Australia. E-mail: keshav.sood@deakin.edu.au.

• Chaoli Sun is with the School of Computer Science and Technology, Taiyuan University of Science and Technology, Taiyuan 030024, China. E-mail: chaoli.sun@tyust.edu.cn.

* To whom correspondence should be addressed.

Manuscript received: 2022-08-11; revised: 2022-09-27; accepted: 2022-10-21

integration of smart technology and traditional power grid that can achieve the goals of economy, reliability, efficiency, and security. During the construction of the SG, a lot of advanced metering infrastructures (AMIs) have been arranged to effectively manage energy usage while obtaining massive power consumption data. Big data and artificial intelligence's emergence have increased interest in DL-based load forecasting techniques[5–7]. Compared with ML method, data-driven DL models no longer rely on expert experience for feature selection and have a stronger adaptive ability and better prediction performance. Owing to the good performance of recurrent neural networks (RNN) on time series data, almost all scholars in the world are working on solving load forecasting problems with RNN and its variants. A model forecasting household load based on deep RNN (DRNN) is proposed[5], which overcomes the problem of over-fitting existing in traditional DL approaches. Kong et al.[6] proposed a long short-term memory (LSTM) approach to tackle the issue of short-term load forecasting. Further, Alhussein et al.[7] suggested a hybrid model called CNN-LSTM, which utilizes the advantages of convolutional neural networks (CNN) in feature extraction and LSTM in the sequence data processing. The framework they developed performed significantly better than competing techniques when tested on real datasets. However, some security and privacy issues have hindered subsequent research and the application of load forecasting models.

Unreasonable and insecure use of data is identified as a major challenge. Smart meters, the centerpiece of AMI, record residential energy consumption and regularly upload these data to energy providers. The powerful predictive potential contained in smart meter data makes it sensitive to personal privacy data. Therefore, consumers generally oppose installing smart meters due to data privacy and security concerns[8]. Referring to the relevant provision in the general data protection regulation (GDPR), the collection or storage of customer electricity usage data is also severely restricted by data minimization principles and consent principles. On the other hand, with SG progressing rapidly, the scale of data collected into smart meters is increasing at an unprecedented rate. Traditional centralized model training is limited by communication and computing power, which makes it difficult to collect and store massive data[4]. Furthermore, when AMI is used to transmit user electricity data, these user data will also be exposed to the risk of data theft[9], data tampering[10], and false data injection[11].

To overcome these issues, federated learning (FL) provides a new solution for SG with distributed edge nodes. In FL[12], the shared global model is trained by the participating devices' federation, which is hosted by the central server. This approach enables edge nodes to cooperate in training models locally without sharing raw training data. Local model parameters, instead of training data, are uploaded to the central server to update the global model. Despite FL's obvious privacy advantages, recent research has indicated that FL underperforms in some areas, such as adversaries still being able to roughly infer sensitive information based on shared parameters throughout the training process[13], and global models of FL training perform poorly for specific users[14]. Given all this, when deploying FL in practice, the resulting system must not only be accurate, but also meet some practical constraints regarding privacy, robustness, and personalization[12].

At present, the privacy and personalization of FL have been studied. The central server of traditional FL is a vulnerable point, so the decentralized FL method based on blockchain[15] is an effective approach to defend against single-point failures and relieve the over-dependence on the central server. Furthermore, there are some privacy-preserving techniques[16] commonly used in FL, such as differential privacy (DP), homomorphic encryption (HE), and secure multi-party computing (SMC). But most of these approaches are server or global model enhancements, and very few work on user-level applications. Some existing efforts, such as the customizable reliable differential privacy (CRDP)[17], take into account user-level privacy concerns, but they are far from being deployed due to added noise and reduced accuracy. In addition, using FL to train neural networks typically runs into problems with not-independent-and-identically-distributed (non-IID) data[14]. The shared global model created by aggregating these various model updates has the potential to slow convergence and cannot be personalized for specific clients.

Based on the challenges discussed above, the key contributions of this article can be categorized as follows.

(1) We propose a user-level load forecasting system based on personalized federated learning (PFL), which can get the global model and personalized model for each client. The personalized model has a better performance than the global model on users' private data, and the global model can play an active role in regional

electricity prediction.

(2) We utilize a novel generative adversarial network (GAN) based DP algorithm (GAN-DP) to protect privacy in our system. The algorithm adds adjustable noise to the local model parameters to meet DP requirements and achieves a tradeoff between privacy protection and prediction accuracy based on GAN theory.

(3) We have conducted extensive experiments on real-world datasets. We evaluate the experimental performance of the personalized model and privacy protection of the load forecasting system. The final results demonstrate that the system performs better than the baseline model, and the application prospect is preliminarily demonstrated.

The remainder of this article is organized as follows. We provide a brief summary of related work in Section 2. Section 3 presents the theoretical approach to our work. We then describe in detail the overall architecture of our load forecasting system and discuss the feasibility of the proposal in Section 4. Next, We provide an evaluation of the performance of our proposal in Section 5. Finally, Section 6 summarizes the entire paper's work.

2 Related Work

2.1 AMI architecture

AMI has always been regarded as an important part of SG, which records customer purchases and transmits these data back to the AMI host system for monitoring and billing. Here, we provide a brief overview of the data transmission procedure in the SG's AMI framework. Figure 1 depicts the AMI architecture, which can be divided into user side, wide area network, and utility side[18]. On the traditional user side, smart meters are the core equipment. Smart meters are mainly responsible for collecting and transmitting customer consumption data within a specific time interval. For the past few years, the growth of Internet of things (IoT) devices, such as smart homes, smartphones, and personal computers, has also broadened the functional extension of the user side, making it possible to realize more operations on user electricity consumption data. The wide area network is mainly responsible for the implementation of bidirectional communication between the utility system and the customer domain. On the utility side, a centralized meter data management system (MDMS) surrounded by master operations and management services acts as the control center for the AMI to manage, store, and analyze customer consumption data.

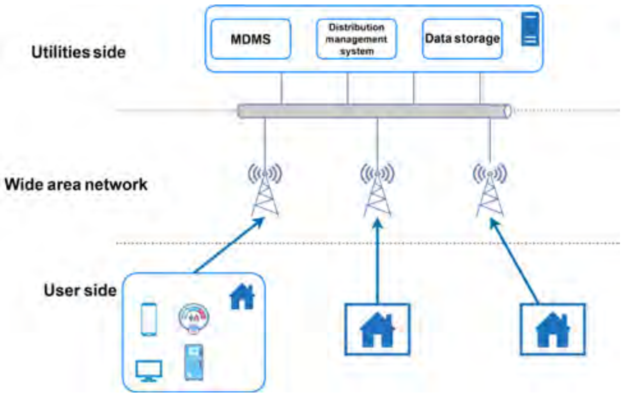


Fig. 1 AMI architecture.

The main participants on the user side are consumers with smart meters installed in their homes, who can use home IoT devices to train local models and upload parameters to the central server. We combine smart meters with these IoT devices as an edge node that has data collection and some data processing capability. AMI's distributed network architecture and the hardware capabilities of edge nodes make it possible to deploy FL.

2.2 FL and some challenges

FL is the latest advancement in distributed ML without sharing clients' private data. While user data are kept locally to protect them from eavesdropping by hidden adversaries, FL also faces some challenges, such as data security issues, disclosure of private information, and poor model performance due to non-IID. Here, we provide a quick summary of the advances in FL's privacy, personalization, and the present work on load forecasting using FL.

Privacy research in FL. Researchers have proposed several compelling solutions[19–21] to solve the issue of privacy leakage during FL and applied them to real-world scenarios. The existing privacy protection methods are mainly developed from three basic technologies: SMC, HE, and DP. SMC[19] is the algorithmic protocol for privacy computing based on cryptography, which can be regarded as the comprehensive application of a variety of cryptography basic tools. HE[20] is a ciphertext computing solution that does not require decryption of the ciphertext. It allows addition and multiplication of the ciphertext but encrypts only a single bit. Although these two can achieve high privacy and accurate calculation results, they also have relatively high requirements for computing and communication capacity. DP[21] defines a strict privacy protection model, which adds a certain distribution of random noise to the dataset to be processed and then obtains a new disturbed dataset to achieve the purpose of data privacy protection. At the same time, DP provides a strict mathematical proof and quantitative evaluation for privacy protection level. Compared with the former two, DP occupies less computing power and can obtain similar privacy performance, which better matches the hardware basis of edge nodes in AMI. However, the problem of data utility reduction caused by adding noise needs to be solved.

Personalization in FL. Existing FL research focuses on training a single global model that can only gain the common characteristics of clients involved in training, and it may perform poorly on specific users. To overcome these problems, PFL intends to provide personalized models for each client in the federation. Significant research is ongoing in this PFL's direction. Li and Wang[22] proposed FedMD, which enables clients to train independent models using their local data based on transfer learning. Each client first performs transfer learning by training the model on a common dataset and then fine-tuning on local data before the FL training phase. Wang et al.[23] proposed a PFL framework called FAVOR, which chooses a portion of clients involved in training for each round to reduce the deviation brought by non-IID data. Arivazhagan et al.[24] devised a basic+personalization layer for deep feedforward neural networks. The base layer is shared with the central server, and the personalization layer remains private on the client side for local training.

FL-based load forecasting. At present, there are many load forecasting works based on FL, and some progress has been made in different research directions. Venkataramanan et al.[25] applied FL to predict distributed energy resources and achieve good forecasting performance in actual grid services such as load swings and load curtailments. The private property of electricity consumption data also attracts people's attention to its data security and privacy protection. Qureshi et al.[26] demonstrated the feasibility of using poison attack to attack the FL-based load forecasting system. Therefore, Sun et al.[27] proposed an FL model based on improved DP algorithm to enhance the privacy protection performance of the load forecasting system. Meanwhile, during the study, the researchers realized that the diversity of load data distribution and load patterns might affect the accuracy of a single global model created by FL. To solve this problem, Gholizadeh and Musilek[28] proposed a new consumer clustering technique using FL, which can better reflect consumers' consumption patterns. Wang et al.[29] went a step further and proposed a personalized federated method for user load forecasting. Specifically, a group of consumers first jointly train the prediction model on a shared smart meter data pool, and then each consumer personalizes the federated prediction model based on their data. All the above studies only strengthen a single attribute but do not comprehensively consider the privacy, personalization, and other issues in the actual load forecasting scenario, and ignore the interaction of these attributes.

2.3 Brief introduction of GAN

As a powerful DL method, GAN has been widely used in various unstructured data fields, including image learning, data processing, and text mining[30–32]. GAN can generate fake data samples that approximate the real data without depending on any data distribution. The structure of GAN mainly consists of two parts: generator and discriminator. The basic idea is that the generator generates fake data samples as realistically as possible by learning real data samples, and the discriminator is used to judge whether the generated samples are different from the real samples. If the discriminator identifies the generated false data as true, it will be retained. The misjudgment of generated samples indicates the consistency of generated samples and real samples in feature space in a way. The proposal of GAN breaks through the problems existing in the previous generative model and improves the generalization ability of the model.

3 Methodology

Here, we introduce the methodologies to build the user-level load forecasting system. The methodologies adopted in our work include multi-task FL and a GAN-based DP algorithm.

3.1 Multi-task FL through personalization

FL allows multiple clients to collaboratively train a global model. In general, the global objective is to address

$$\min_{\omega} f(\omega) = \sum_{k=1}^{N} p_k F_k(\omega) \tag{1}$$

where N is the number of devices and $F_k(\omega)$ is the local objective for device k. In FedAvg setting, p_k is a non-negative weight value, we can set $p_k = \frac{n_k}{n}$, $\sum_{k=1}^{N} p_k =$

1, where n_k samples available at each device k, and $n = \sum_{k=1}^{N} n_k$ is the total number of data points.

However, in fact, local data x_k on different devices present different distributions $\mathcal{D}_k$, i.e., $F_k(\omega) := E_{x_k \sim \mathcal{D}_k}[f_k(\omega; x_k)]$, and the global model trained by FL has poor performance for these clients. To solve the data heterogeneity, it is essential to get a personalized model $\{v_k\}_{k \in [N]}$ for each client. Here we utilize Ditto[33], a multi-task FL framework, to customize the personalization model. The bi-level optimization problem between the global objective and the local objective can be formulated as

$$\min_{v_k} h_k(v_k; \omega^*) := F_k(v_k) + \frac{\lambda}{2}\|v_k - \omega^*\|^2, \\ \text{s.t.} \quad \omega^* \in \arg\min_{\omega} f(\omega) \tag{2}$$

where the hyperparameter λ adjusts the interpolation between global and local models. In particular, we note that the global model can be gotten with $\lambda \to +\infty$, and Ditto will be much closer to training the local models as $\lambda \to 0$.

As mentioned above, we jointly train the global model ω^* and personalized models $\{v_k\}_{k \in [N]}$ in an alternate manner in Ditto. For client $k \in S_t$, S_t is the number of equipments involved in training in this iteration. With their private data, each client involved in the training will train a local model at t iteration.

$$\omega_k^{t+1} \leftarrow \omega^t - \eta_g \nabla F_k(\omega_k^t) \tag{3}$$

In the parameter update section, personalized model parameters v_k^{t+1} are updated in parallel via the global-regularized approach.

$$v_k^{t+1} = v_k^t - \eta_l \left(\nabla F_k\left(v_k^t\right) + \lambda\left(v_k^t - w^t\right)\right) \tag{4}$$

where η_g and η_l correspond to different learning rates when updating the global model and the personalized model, respectively. At the same phase, here we use FedAvg (or other optimization strategies) for the global model parameters ω^{t+1} update.

$$w^{t+1} \leftarrow w^t + \frac{1}{|S_t|}\sum_{k \in S_t}\left(\omega_k^{t+1} - \omega_k^t\right) \tag{5}$$

This iterative process is repeated until convergence or a preset value of training rounds is reached, and we end up with the global models ω^* and personalized models $\{v_k\}_{k \in [N]}$ for each client.

3.2 GAN-DP modeling for PFL

Using DP algorithm in FL is an effective way to obtain a strong privacy guarantee. However, DP's sacrifice of accuracy hinders its entry into practical application scenarios. Here we adopt GAN-DP[34], a modified GAN model, in the above FL setting, which improves the load forecasting accuracy while complying with the DP requirements.

As shown in Fig. 2, GAN-DP contains a generator, a discriminator, and the DP identifier (DPI) whose structure is similar to the discriminator. In our system, the local parameters are obtained through local training by the client. Then, the local model parameters are fed into the generator, and the generator produces a group of synthesized parameters. The generated synthesized parameters are then treated as inputs to the discriminators and the DPI. If the synthesized parameter satisfies the requirements of the two perceptrons, the parameter is taken as the output result and goes to the next step. Table 1 provides brief instructions for notations in this chapter.

Generator: Utilizing the original local model parameters, the generator produces synthesized parameters and submits them to the discriminator for identification. The N_{d_i} noise samples $\{y_1, y_2, \ldots, y_n\}$ from $p_g(y)$ are inputted and updated by

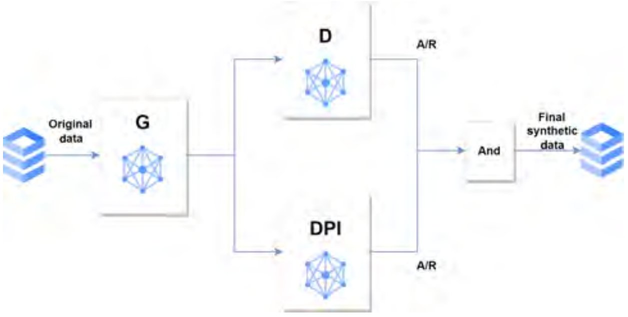


Fig. 2 GAN-DP. "A" and "R" indicate acceptance and rejection, respectively.

Table 1 Notations in GAN-DP.

Notation	Explanation
d_i	Training data
N_{d_i}	Number of noise samples
p_g	Distribution of generator
$\delta(d_i)$	Data samples for training
$E(\cdot)$	Mathematical expectation
$p_g(y)$	Prior injected noise
$G(y; \theta_g)$	Multilayer perceptron of generator
$D(y; \theta_d)$	Multilayer perceptron of discriminator
$I(y; \theta_i)$	Multilayer perceptron of DP identifier
$S(D; I)$	Combined structure of the discriminator and DP identifier

$$\nabla_{\theta_g}\frac{1}{N_{d_i}}\sum_{i=1}^{N_{d_i}}\log\left(1-D\left(G\left(y_i\right)\right)\right) \tag{6}$$

Discriminator: After several iterations, if the discriminator identifies the synthesized parameters as the original model parameters, the final result is obtained. Here we select N_{d_i} data samples $\{d_1, d_2, \ldots, d_{N_{d_i}}\}$ from $\delta(d_i)$, and the gradient ascent of discriminator in the update process can be formulated by

$$\nabla_{\theta_d}\frac{1}{N_{d_i}}\sum_{i=1}^{N_{d_i}}\left[\log D\left(d_i\right)+\log\left(1-D\left(G\left(y_i\right)\right)\right)\right] \tag{7}$$

DP identifier: Since the DPI serves as a discriminator, the discriminator and DPI interact with the generator in parallel. Unlike the discriminator, DPI attempts to confirm whether the synthesized model parameters satisfy the DP requirements. The update procedure can be expressed as

$$\nabla_{\theta_{N_{d_i}}}\frac{1}{N_{d_i}}\sum_{i=1}^{N_{d_i}}\times$$
$$\left[\log S\left(d_i \mid D; I\right)+\log\left(1-S\left(G\left(y_i\right) \mid D; I\right)\right)\right] \tag{8}$$

In GAN-DP, there is a min-max game established between the generator, discriminator, and DPI. Based on the above formulas, this problem can be modeled as

$$\min_G \max_S E_{x\sim\delta(d_i)}\left[\log S\left(d_i \mid D; I\right)\right]+$$
$$E_{y\sim p_g(y)}\left[\log\left(1-S\left(G\left(y_i\right) \mid D; I\right)\right)\right] \tag{9}$$

In Formula (9), we use $\min_G$ to reduce the likelihood of discrimination and $\max_S$ to increase the likelihood of deception.

The PFL for heterogeneous residential load forecasting is shown in Algorithm 1.

Algorithm 1 PFL for heterogeneous residential load forecasting

```
Parameters initialization;
for number of training iterations do
    Server randomly selects a subset $S_t$ of $N$ devices;
    Server sends $\omega^t$ to all chosen devices;
    for chosen devices in parallel do
        local model training using Formula (3);
        synthesized local model parameters using Formula (9);
        personalized model parameters update using Eq. (4);
        send global update;
    end
    Server aggregates using Formula (5);
end
return personalized models for each client; global model.
```

4 System Modelling

In this section, a user-level load forecasting system based on PFL is introduced. Here we ignore the effect of communication delay to simplify the system. Finally, we discuss the feasibility of the system.

4.1 System architecture

Figure 3 depicts the overall system's architecture in our work. From Fig. 3, it is clear that the whole system is made up mostly of two components: the central server and the clients. In load forecasting scenarios, the power company is the main body of the central server, and it relies on the SG for power transmission and management. Residential customers, defined as clients here, have smart meters to measure electricity consumption, train local models, and communicate with the central server. Here we conclude the specific

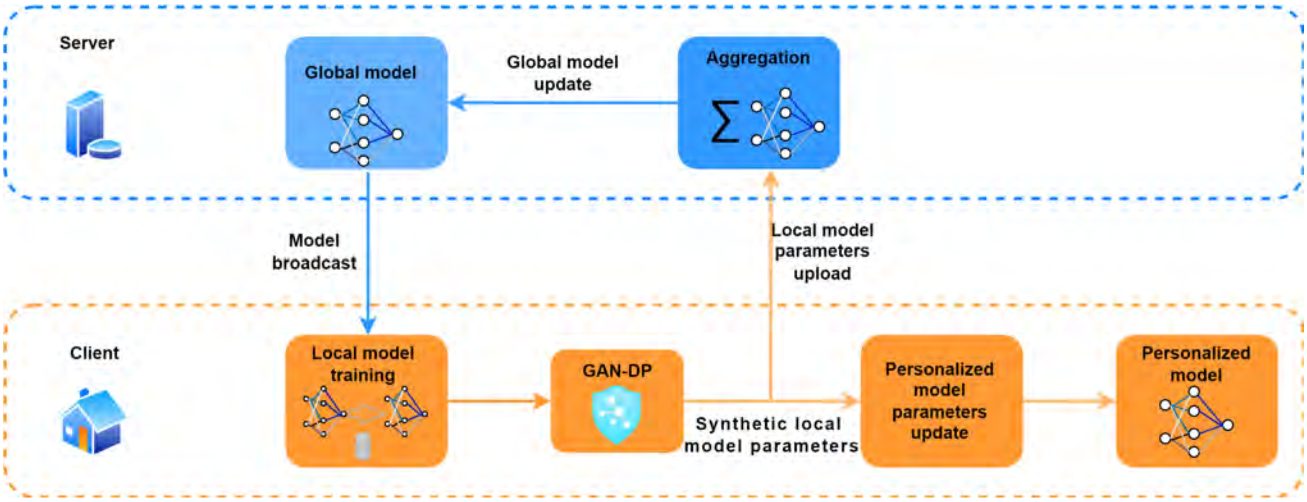


Fig. 3 System architecture.

procedures of the proposed system as follows.

Global model initialization: To begin training, each client receives the initial weight values of the global model from the central server. Here the weight values are initialized by assigning random values.

Local model training: After the parameters are updated with the downloaded weight values, the smart meter trains a local model using the client's local private data.

Synthesizing local model parameters: The generator of GAN-DP receives the trained local model parameters and outputs a set of synthesized parameters. The synthesized data are then injected into the discriminator and the DPI to see if they meet their requirements. Once the requirements are met, the synthesized data are treated as local model parameters to participate in subsequent steps.

Personalized model update: Here we follow the Ditto setting to update the personalized model parameters for each client.

Local model parameters upload: Each client participating in the training uploads synthesized parameters to the central server.

Global aggregation: New global model parameters are produced via aggregating the model updates uploaded in the central server.

Global model update: The global model performs weight updates using the parameters generated from the aggregator.

Model broadcast: The updated weights are broadcasted back to the clients by the central server for the next round of training.

This iteration continues until the respective models (both global and personalized) converge.

4.2 Feasibility discussion

Let us first discuss the feasibility of deploying the proposed load forecasting system in the real world. The computing and communication capabilities of user devices are essential for deploying an FL framework. A combination of smart meters and other smart devices such as smartphones can serve as a node for FL needs. In fact, it is feasible for smart meters to communicate directly with the cloud using existing technology[35]. In addition, we use Ditto for the personalized forecasting model due to its various benefits. Compared with other PFL, Ditto is smaller and more effective. And more importantly, Ditto is especially helpful in real-world scenarios when we must simultaneously consider several constraints (accuracy, fairness, and robustness). Because of its multi-task learning structure design, we can also obtain the global model and the personalized model of each user at the same time. The prediction accuracy of the personalized model in user electricity data is better than that of the global model. Although the global model does not perform well for individual users, we find that the global model can be used to forecast regional load (details in Section 5), which broadens the original capability. Our prediction system further improves the prediction accuracy while preserving privacy via GAN-DP. Here we only use the basic LSTM because the prediction model is not our focus. We can introduce the state-of-the-art LSTM model to further improve the prediction accuracy in the future.

5 Performance Evaluation

This section describes the details of the dataset, the simulation platform, and the parameters of the model used in our work. After that, it presents and evaluates the experimental results.

5.1 Data analysis and pre-processing

The dataset used in the experiments is the "HUE[36]: The Hourly Usage of Energy Dataset for Buildings in British Columbia" dataset. There are currently 22 households in the dataset, most with a 3-year history of energy consumption. To obtain as much data as possible for the same period of electricity consumption, we limited the period between "2015-08-21" and "2018-01-29". Based on this rule, houses with IDs 3–6, 8–14, and 18–20 were selected for a total of 14 families.

After selecting the appropriate data, we need to preprocess the data. In this paper, the missing value of the dataset is processed according to the mean or interpolation near the missing value, which is formally described as

$$x_i = \begin{cases} \frac{x_{i-1}+x_{i+1}}{2}, & x_i \in \text{Null}, x_{i-1}, x_{i+1} \notin \text{Null}; \\ 0, & x_i \in \text{Null}, x_{i-1} \text{ or } x_{i+1} \in \text{Null}; \\ x_i, & x_i \notin \text{Null} \end{cases} \tag{10}$$

where x_i represents the power consumption data over a period.

Sequentially, we need to normalize the power consumption data to smoothen the convergence. Here we choose the max-min scaling method for normalization.

Finally, the processed data were partitioned into training, validation, and testing datasets using a $0.6/0.2/0.2$ split.

5.2 Simulation setting

The proposed system is implemented on a workstation with an Intel(R) Xeon(R) W-2255 CPU, NVIDIA GeForce RTX 3090 GPU (20 cores), and 32 GB RAM. The case study is operated on FederatedScope[37], a comprehensive FL platform developed by Alibaba Group, with Python 3.9.

Since LSTM is convenient for sequence modeling and has the ability of long-term memory, we use LSTM as a prediction model in our system. To satisfy the input requirements of the LSTM model, the time-series sequences need to be processed by sliding windows. Here we set the size of sliding windows $t = 24\,\text{h}$: this means that 24-h continuous electricity consumption data are fed into the LSTM model and the model outputs the predicted value of electricity consumption in the next hour. The network applied in our system contains two hidden LSTM layers and two fully connected layers. Other hyperparameter settings of the experiment can be shown in Table 2.

Mean square error (MSE) and root mean square error (RMSE) are typical evaluation criteria in regression problems, so we use MSE as the loss function and RMSE as the evaluation criterion in our experiments. The smaller the MSE and RMSE value, the better the model performance.

$$\text{MSE} = \frac{\sum\limits_{i=1}^{M} (y_i - \hat{y}_i)^2}{M} \tag{11}$$

$$\text{RMSE} = \sqrt{\frac{\sum\limits_{i=1}^{M} (y_i - \hat{y}_i)^2}{M}} \tag{12}$$

where y_i is the actual energy consumption measurement, $\hat{y}_i$ is the predicted value, and M represents the total number of predicted values.

Table 2 Hyperparameter setting.

Hyperparameter	Value
Number of epochs of training on clients	5
Total communication round	500
Fraction of clients participating	0.3
Model structure	LSTM layer with 128 hidden states LSTM layer with 256 hidden states Fully connected layers with 64 neurons Fully connected layers with 32 neurons
Privacy budget ε	2, 4, 6, 8, 10
Loss	MSE
Batch size	128
Learning rate	0.0001

5.3 Result analysis

To demonstrate the superiority of our suggested approach, we contrast the prediction performance of different models on this dataset. These models include the traditional global model using FedAvg (Global), PFL with Ditto framework only (PFL), PFL with DP algorithm (PFL-DP), and our proposed system (PFL-GANDP).

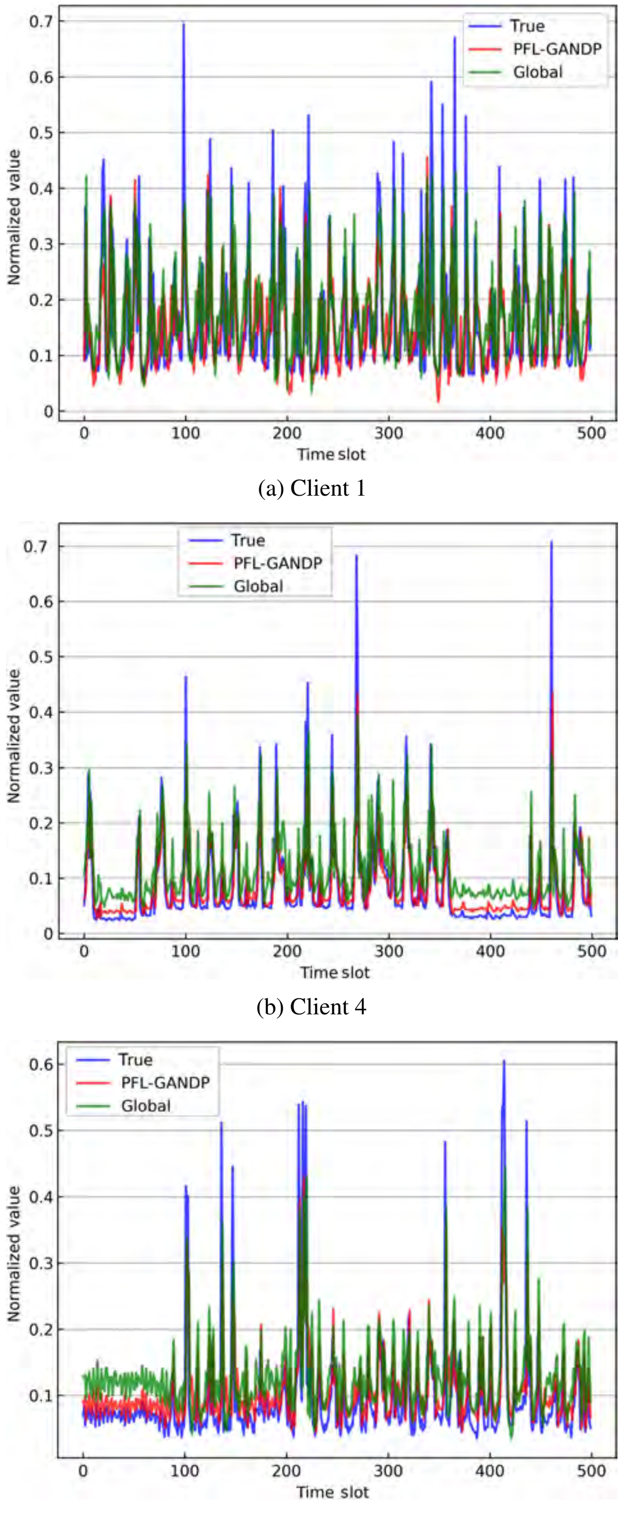
Table 3 shows the test losses of these benchmark models on 14 customers power consumption data. Here we adopt DP based on the Laplace mechanism and set $\varepsilon = 6$. The success of the personalization strategy in the area of predicting user load is evident from Table 3. PFL outperforms Global in predicting specific users. Due to the addition of noise, the prediction performance of PFL, PFL-GANDP, and PFL-DP showed a downward trend. These variances can be visualized in the pictures shown later.

5.3.1 Performance of personalized load forecasting

We compared the performance of these models using test datasets for three randomly selected customers. Figure 4 illustrates forecasting results from the global model and PFL-GANDP model of clients 1, 4, and 7. In contrast, our proposed model's prediction value is more in line with the actual value. Although the global model does not perform well for individual users, we find new scenarios where the global model is applicable. We integrate the data of these 14 customers into a regional

Table 3 Test loss (RMSE) for 14 customers using different models.

House ID	RMSE			
	Global	PFL	PFL-DP	PFL-GANDP
3	0.1991	0.1123	0.2388	0.2033
4	0.1406	0.0813	0.1930	0.1342
5	0.1465	0.0806	0.2251	0.1335
6	0.1046	0.0595	0.1774	0.1114
8	0.1532	0.0867	0.1832	0.1546
9	0.1192	0.0665	0.1875	0.1069
10	0.1338	0.0777	0.1788	0.1259
11	0.1345	0.0766	0.2042	0.1278
12	0.0926	0.0528	0.1583	0.1034
13	0.1466	0.0847	0.1890	0.1370
14	0.1077	0.0628	0.1443	0.1169
18	0.1821	0.1043	0.2679	0.1655
19	0.1593	0.0896	0.1997	0.1684
20	0.1410	0.0791	0.2312	0.1312



(a) Client 1

(b) Client 4

(c) Client 7

Fig. 4 Forecasting results for electrical load using global and our proposed model.

electricity consumption dataset by time and conduct training on this dataset. And to our surprise, the global model worked well on regional consumption data, and the forecasting results are presented in Fig. 5. This means that our system, which integrates global and personalized models, can better serve load forecasting scenarios.

5.3.2 Comparison with a traditional DP strategy

Furthermore, we conducted additional experiments to demonstrate our proposed system's superiority in privacy protection. In Fig. 6, we compare the prediction performance of PFL-GANDP and PFL-DP for different privacy budgets ε. Horizontally, with the privacy budget increases, the prediction accuracy of the two models also improves. We know that the smaller the privacy budget setting in DP, the larger the added noise, the better the privacy protection, and the worse the prediction accuracy. Through vertical comparison, under the same privacy budget, our proposed system can achieve better prediction performance than PFL-DP, which means that GAN-DP can achieve a better trade-off between privacy protection and prediction performance. Figure 7

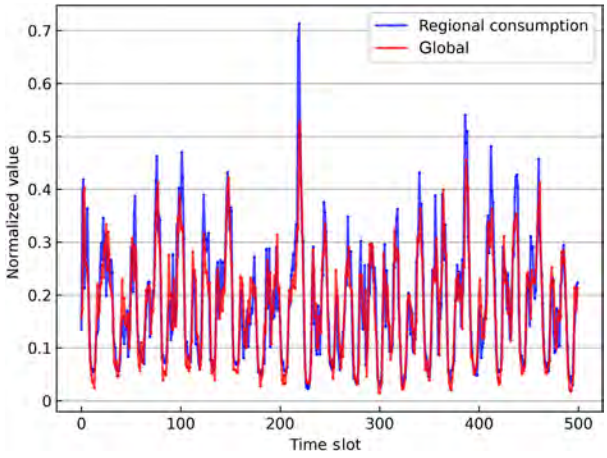


Fig. 5 Forecasting results for electrical load on regional consumption using global model.

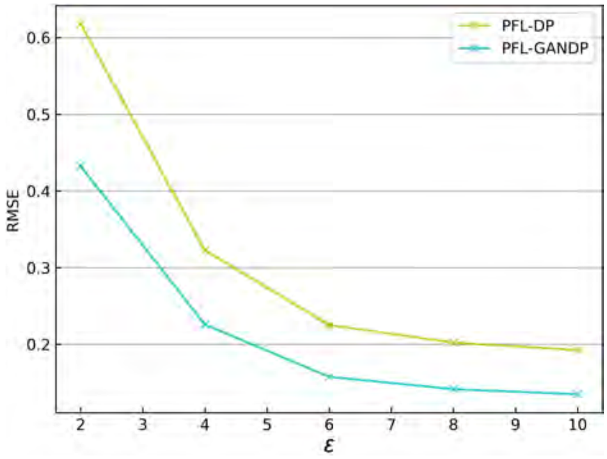


Fig. 6 Comparison of privacy protection effect.

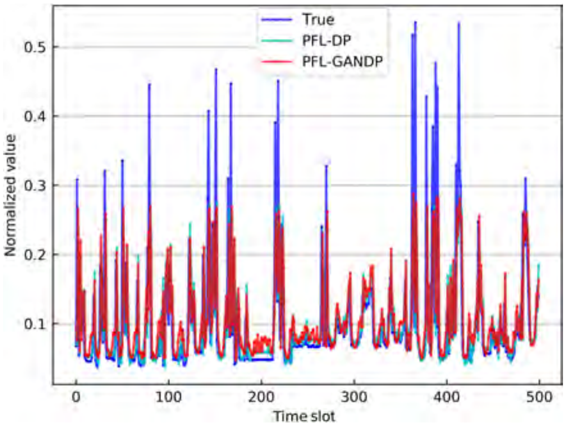


Fig. 7 Forecasting performance on privacy protection strategy.

provides a visual representation of the prediction performance of these two models on the user dataset. Note that all models can provide privacy protection by deviating from the solid blue line (real data).

6 Conclusion

In this research, we propose a system for user-level load forecasting based on PFL. We also find that the global model obtained simultaneously can effectively predict the regional electricity consumption data, which broadens the application scenarios of our load forecasting system. We further apply GAN-DP to increase the system's privacy protection while minimizing the impact on prediction accuracy. The results of our experiments showed that our system could forecast user-level load information with accuracy and privacy protection.

We intend to expand our current work in the future according to the demands of the actual load forecasting scenario. Considering that terminal devices with different performances may not be able to upload their local model parameters synchronously in real scenarios, we will try to introduce the idea of asynchronous FL to solve the latency problem. In addition, the private nature of user electricity data is attractive to malicious people, and the distributed nature and data constraints of FL open up new failure modes and attack surfaces for attackers. So we intend to investigate the attack and defense strategies under load forecasting to improve the robustness of the whole system.

Acknowledgment

This work was supported by the Scientific and Technologial Innovation Programs of Higher Education Institutions in Shanxi, China (No. 2020L0338) and the Shanxi Key Research and Development Program (Nos. 202102020101002 and 202102020101005).

References

[1] S. Haben, S. Arora, G. Giasemidis, M. Voss, and D. V. Greetham, Review of low voltage load forecasting: Methods, applications, and recommendations, *Applied Energy*, vol. 304, p. 117798, 2021.

[2] S. Dong, P. Wang, and K. Abbas, A survey on deep learning and its applications, *Computer Science Review*, vol. 40, p. 100379, 2021.

[3] M. Injadat, A. Moubayed, A. B. Nassif, and A. Shami, Machine learning towards intelligent systems: Applications, challenges, and opportunities, *Artificial Intelligence Review*, vol. 54, no. 5, pp. 3299–3348, 2021.

[4] M. Faheem, S. B. H. Shah, R. A. Butt, B. Raza, M. Anwar, M. W. Ashraf, M. A. Ngadi, and V. C. Gungor, Smart grid communication and information technologies in the perspective of industry 4.0: Opportunities and challenges, *Computer Science Review*, vol. 30, pp. 1–30, 2018.

[5] H. Shi, M. Xu, and R. Li, Deep learning for household load forecasting—A novel pooling deep RNN, *IEEE Transactions on Smart Grid*, vol. 9, no. 5, pp. 5271–5280, 2017.

[6] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, Short-term residential load forecasting based on LSTM recurrent neural network, *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 841–851, 2017.

[7] M. Alhussein, K. Aurangzeb, and S. I. Haider, Hybrid CNN-LSTM model for short-term individual household load forecasting, *IEEE Access*, vol. 8, pp. 180544–180557, 2020.

[8] N. Balta-Ozkan, O. Amerighi, and B. Boteler, A comparison of consumer perceptions towards smart homes in the UK, Germany and Italy: Reflections for policy and future research, *Technology Analysis & Strategic Management*, vol. 26, no. 10, pp. 1176–1195, 2014.

[9] A. Jindal, A. Dua, K. Kaur, M. Singh, N. Kumar, and S. Mishra, Decision tree and SVM-based data analytics for theft detection in smart grid, *IEEE Transactions on Industrial Informatics*, vol. 12, no. 3, pp. 1005–1016, 2016.

[10] M. G. Chuwa and F. Wang, A review of non-technical loss attack models and detection methods in the smart grid, *Electric Power Systems Research*, vol. 199, p. 107415, 2021.

[11] L. Cui, Y. Qu, L. Gao, G. Xie, and S. Yu, Detecting false data attacks using machine learning techniques in smart grid: A survey, *Journal of Network and Computer Applications*, vol. 170, p. 102808, 2020.

[12] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al., Advances and open problems in federated learning, arXiv preprint arXiv: 1912.04977, 2019.

[13] H. Lee, J. Kim, R. Hussain, S. Cho, and J. Son, On defensive neural networks against inference attack in federated learning, in *Proc. 2021 IEEE International Conference on Communications*, Montreal, Canada, 2021, pp. 1–6.

[14] H. Zhu, J. Xu, S. Liu, and Y. Jin, Federated learning on non-IID data: A survey, *Neurocomputing*, vol. 465, pp. 371–390, 2021.

[15] C. Li, Y. Yuan, and F. Y. Wang, Blockchain-enabled federated learning: A survey, in *Proc. 2021 IEEE 1st International Conference on Digital Twins and Parallel Intelligence* (*DTPI*), Beijing, China, 2021, pp. 286–289.

[16] X. Yin, Y. Zhu, and J. Hu, A comprehensive survey of privacy-preserving federated learning: A taxonomy, review, and future directions, *ACM Computing Surveys* (*CSUR*), vol. 54, no. 6, pp. 1–36, 2022.

[17] Y. Qu, S. Yu, W. Zhou, S. Chen, and J. Wu, Customizable reliable privacy-preserving data sharing in cyber-physical social networks, *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 1, pp. 269–281, 2020.

[18] X. Fang, S. Misra, G. Xue, and D. Yang, Smart grid—The new and improved power grid: A survey, *IEEE Communications Surveys & Tutorials*, vol. 14, no. 4, pp. 944–980, 2011.

[19] K. Sahinbas and F. O. Catak, Secure multi-party computation based privacy preserving data analysis in healthcare IoT systems, arXiv preprint arXiv: 2109.14334, 2021.

[20] L. Zhang, J. Xu, P. Vijayakumar, P. K. Sharma, and U. Ghosh, Homomorphic encryption-based privacy-preserving federated learning in IoT-enabled healthcare system, *IEEE Transactions on Network Science and Engineering*, doi: 10.1109/TNSE.2022.3185327.

[21] B. Jiang, J. Li, H. Wang, and H. Song, Privacy-preserving federated learning for industrial edge computing via hybrid differential privacy and adaptive compression, *IEEE Transactions on Industrial Informatics*, vol. 19, no. 2, pp. 1136–1144, 2021.

[22] D. Li and J. Wang, FedMD: Heterogenous federated learning via model distillation, arXiv preprint arXiv: 1910.03581, 2019.

[23] H. Wang, Z. Kaplan, D. Niu, and B. Li, Optimizing federated learning on non-IID data with reinforcement learning, in *Proc. IEEE INFOCOM 2020-IEEE Conference on Computer Communications*, Toronto, Canada, 2020, pp. 1698–1707.

[24] M. G. Arivazhagan, V. Aggarwal, A. K. Singh, and S. Choudhary, Federated learning with personalization layers, arXiv preprint arXiv: 1912.00818, 2019.

[25] V. Venkataramanan, S. Kaza, and A. M. Annaswamy, DER forecast using privacy-preserving federated learning, *IEEE Internet of Things Journal*, vol. 10, no. 3, pp. 2046–2055, 2022.

[26] N. B. S. Qureshi, D. H. Kim, J. Lee, and E. K. Lee, Poisoning attacks against federated learning in load forecasting of smart energy, in *Proc. 2022 IEEE/IFIP Network Operations and Management Symposium*, Budapest, Hungary, 2022, pp. 1–7.

[27] M. Sun, J. Li, Y. Ren, S. Fang, and J. Yan, Research on federated learning and its security issues for load forecasting, in *Proc. 2021 13th International Conference on Computer Modeling and Simulation*, Melbourne, Australia, 2021, pp. 237–243.

[28] N. Gholizadeh and P. Musilek, Federated learning with hyperparameter-based clustering for electrical load forecasting, *Internet of Things*, vol. 17, p. 100470, 2022.

[29] Y. Wang, N. Gao, and G. Hug, Personalized federated learning for individual consumer load forecasting, *CSEE Journal of Power and Energy Systems*, doi: 10.17775/CSEEJPES.2021.07350.

[30] D. Li, X. Nie, X. Li, Y. Zhang, and Y. Yin, Context-related video anomaly detection via generative adversarial network, *Pattern Recognition Letters*, vol. 156, pp. 183–189, 2022.

[31] Y. Lu, D. Chen, E. Olaniyi, and Y. Huang, Generative adversarial networks (GANs) for image augmentation in agriculture: A systematic review, *Computers and Electronics in Agriculture*, vol. 200, p. 107208, 2022.

[32] A. Wali, Z. Alamgir, S. Karim, A. Fawaz, M. B. Ali, M. Adan, and M. Mujtaba, Generative adversarial networks for speech processing: A review, *Computer Speech & Language*, vol. 72, p. 101308, 2022.

[33] T. Li, S. Hu, A. Beirami, and V. Smith, Ditto: Fair and robust federated learning through personalization, in *Proc. 38th International Conference on Machine Learning*, Virtual, 2021, pp. 6357–6368.

[34] L. Cui, Y. Qu, G. Xie, D. Zeng, R. Li, S. Shen, and S. Yu, Security and privacy-enhanced federated learning for anomaly detection in IoT infrastructures, *IEEE Transactions on Industrial Informatics*, vol. 18, no. 5, pp. 3492–3500, 2021.

[35] M. Pau, E. Patti, L. Barbierato, A. Estebsari, E. Pons, F. Ponci, and A. Monti, A cloud-based smart metering infrastructure for distribution grid services and automation, *Sustainable Energy, Grids and Networks*, vol. 15, pp. 14–25, 2018.

[36] S. Makonin, HUE: The hourly usage of energy dataset for buildings in British Columbia, https://doi.org/10.7910/DVN/N3HGRN, 2018.

[37] Y. Xie, Z. Wang, D. Gao, D. Chen, L. Yao, W. Kuang, Y. Li, B. Ding, and J. Zhou, FederatedScope: A flexible federated learning platform for heterogeneity, arXiv preprint arXiv: 2204.05011, 2022.

Xiaodong Qu received the BEng degree in information science and technology from Donghua University, Shanghai, China in 2015. He is currently pursuing the master degree at the Shanxi Key Laboratory of Advanced Control and Equipment Intelligence, Taiyuan University of Science and Technology, Taiyuan, China. His research interests include load forecasting and federated learning.

Chengcheng Guan received the BEng degree from Tongji Zhejiang College, China in 2021. He is currently pursuing the master degree at the Shanxi Key Laboratory of Advanced Control and Equipment Intelligence, Taiyuan University of Science and Technology, Taiyuan, China. His research interests include the strategy of attack and defense in federated learning.

Gang Xie received the PhD degree in circuits and systems from Taiyuan University of Technology, China in 2006. He is currently the vice president of Taiyuan University of Science and Technology, China. He has also been a professor of Taiyuan University of Technology since 2008. He has authored over 100 papers and held ten invention patents, and published one academic monograph and prepared five textbooks. His main research interests include intelligent information processing, big data, and complex networks. He has received six provincial science and technology awards.

Zhiyi Tian received the BS degree in information security from Sichuan University, China in 2017. He is currently pursuing the PhD degree with the Faculty of Engineering and Information Technology, University of Technology Sydney, Australia. His research interests include the security and privacy issues in deep learning and federated learning.

Lei Cui received the PhD degree from Deakin University, Melbourne, Australia in 2021. He has authored or coauthored more than 30 publications, including monographs, book chapters, and journal and conference papers. Some of his publications have been published in the top venues such as *IEEE TII*, *IEEE TNSM*, and *IEEE TPDS*. His research interests include security and privacy issues in IoT, social networks, and machine learning. He is active in communication society and has served as a reviewer for many Q1 journals and a TPC member for international conferences.

Chaoli Sun received the BSc and MSc degrees from Hohai University, Nanjing, China in 2000 and 2003, respectively, and the PhD degree from Taiyuan University of Science and Technology, Taiyuan, China in 2011. She is a professor of the School of Computer Science and Technology, Taiyuan University of Science and Technology. Her research interests include swarm intelligence, surrogate-assisted evolutionary optimization, and their applications to practical engineering problems. She is an associate editor of the *IEEE Transactions on Evolutionary Computation*, the *IEEE Transactions on Artificial Intelligence*, and the *Soft Computing Journal*. She is also an editorial board member of *Complex and Intelligence Systems* and *Memetic Computing*. She was the chair of TF on Data-Driven Evolutionary Optimization of Expensive Problems (2015–2020). She has also served as a reviewer for journals including *IEEE Transactions on Cybernetics*, *IEEE Transactions on Neural Networks and Learning Systems*, and *IEEE Computational Intelligence Magazine*.

Keshav Sood is currently a lecturer with the Centre for Cyber Security Research and Innovation, the School of IT, Deakin University, Australia. Previously, he worked as a research fellow with the Advanced Cyber Security Engineering Research Centre (ACSRC), University of Newcastle, Australia.