energies

MDPI

Article

An IHPO-WNN-Based Federated Learning System for Area-Wide Power Load Forecasting Considering Data Security Protection

Bujin Shi [1], Xinbo Zhou [2,*], Peilin Li [1], Wenyu Ma [3] and Nan Pan [3]

[1] Kunming Power Supply Bureau, Yunnan Power Grid Co., Ltd., Kunming 650011, China; shibujin@km.yn.csg (B.S.); lipeilin@km.kfzx.yn.csg.cn (P.L.)
[2] Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China
[3] Faculty of Civil Aviation and Aeronautics, Kunming University of Science and Technology, Kunming 650500, China; mawenyucjr@outlook.com (W.M.); nanpan@kust.edu.cn (N.P.)
* Correspondence: zhouxinbo@stu.kust.edu.cn; Tel.: +86-153-6887-6179

Abstract: With the rapid growth of power demand and the advancement of new power system intelligence, smart energy measurement system data quality and security are also facing the influence of diversified factors. To solve the series of problems such as low data prediction efficiency, poor security perception, and "data islands" of the new power system, this paper proposes a federated learning system based on the Improved Hunter–Prey Optimizer Optimized Wavelet Neural Network (IHPO-WNN) for the whole-domain power load prediction. An improved HPO algorithm based on Sine chaotic mapping, dynamic boundaries, and a parallel search mechanism is first proposed to improve the prediction and generalization ability of wavelet neural network models. Further considering the data privacy in each station area and the potential threat of cyber-attacks, a localized differential privacy-based federated learning architecture for load prediction is designed by using the above IHPO-WNN as a base model. In this paper, the actual dataset of a smart energy measurement master station is selected, and simulation experiments are carried out through MATLAB software to test and examine the performance of IHPO-WNN and the federal learning system, respectively, and the results show that the method proposed in this paper has high prediction accuracy and excellent practical performance.

Keywords: electricity load forecasting; improved hunter–prey optimizer; WNN; federated learning; differential privacy

check for updates

Citation: Shi, B.; Zhou, X.; Li, P.; Ma, W.; Pan, N. An IHPO-WNN-Based Federated Learning System for Area-Wide Power Load Forecasting Considering Data Security Protection. *Energies* **2023**, *16*, 6921. https://doi.org/10.3390/en16196921

Academic Editor: Daniele D. Giusto

Received: 15 September 2023
Revised: 26 September 2023
Accepted: 28 September 2023
Published: 1 October 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

1.1. Background

With the construction of new power systems and the wide application of digital technology, information sharing, diversified new energy devices, and different business scenarios have become the trend of development in the field of intelligent energy measurement, and power load forecasting has become more and more important in power system operation and scheduling. Power load forecasting technology can be categorized into ultra-short-term load forecasting, short-term load forecasting, medium- and long-term load forecasting, etc., according to the period of forecasting [1–3]. Its use can range from real-time scheduling and operation of the power system, supporting the stable supply of electric power and coping with sudden fluctuations, to the formulation of long-term strategies for the long-term planning and development of the power system, and the development of new energy sources. Therefore, accurate data prediction techniques are crucial for stable power system operation, optimal resource deployment, and energy management.

A station in a power system means the power supply range or area of a transformer. However, the power data collected from each station in the power system usually cannot be fully shared due to network security, privacy protection, and laws and regulations, resulting in an obvious data silo problem. Under the premise of ensuring reasonable information sharing, how to protect the privacy of data information of all parties has also become an urgent problem.

1.2. Literature Review

The research in this paper centers on the problem of power load forecasting and data security protection based on federated learning. Common power load forecasting techniques include time series analysis [4], regression analysis [5], neural networks [6], and generative adversarial networks [7]. In addition, as an important development in the field of machine learning, the time series transformer has shown high potential in processing time series data, as well as a strong migration learning capability [8], providing a promising alternative to traditional neural networks [9]. Among them, neural networks have achieved wider application in the field of power load forecasting due to their powerful nonlinear modeling ability and time series information capturing ability, and at the same time, it is more suitable for small and medium scale forecasting tasks due to their fewer hyper-parameters and more concise structure compared to deep learning networks. In [10], a hybrid modeling approach combining a neural network model and stochastic differential equations is proposed to predict the photovoltaic power generation in different seasons, and at the same time, it can output the prediction results under different confidence intervals. In [11], a multi-space collaborative framework for optimal model selection for power load forecasting is proposed, which not only has good forecasting performance but also is not limited by parameter domains, providing a good reference for power system planning.

To improve the performance and parameter tuning efficiency of the prediction model so that it can fit the training data more accurately, existing studies mainly optimize the prediction model through learning rate scheduling, Batch Normalization, parameter initialization, and hyper-parameter tuning. In addition, the hybrid modeling approach is also a powerful way to achieve efficient parameter tuning of the model; for example, the study described in [12] proposed a modeling approach combining a first-principle model and an artificial neural network (ANN), which makes the hybrid prediction model perform significantly better than the original model. Optimization of the neural network model by meta-heuristic algorithms, using the ability of meta-heuristic algorithms to deal with non-convex problems to optimize the weights, hyper-parameters, etc., can make up for the shortcomings of its learning ability and convergence speed [13]—this approach is more flexible, efficient, and more suitable for the flexible, distributed, high-volume wide-area power load forecasting oriented to this paper's application scenarios.

The data involved in power load forecasting is sensitive and private, containing users' power usage information and behavioral patterns. Traditional power load forecasting systems are often based on centralized data collection and processing [14,15], which easily leads to the risk of leakage and misuse of user data during centralized storage and processing. To solve this problem, federated learning has emerged, which is essentially a distributed machine learning framework in which each federated learning participant can effectively protect the privacy of user data by training the model on local devices and sharing only the training parameters of the model, with the raw data not leaving the local area. While early federated learning research focused on distributed optimization problems, in recent years, with the rise of deep learning, federated learning has been gradually applied to a wider range of tasks, including the Internet of Things, cloud servers, and medical diagnosis [16]. For example, a federated learning framework for UAV edge computing based on the Stackelberg game is proposed in the literature [17] to cope with traffic surge events in healthcare infrastructure to improve the benefits of social good, in which the incentive mechanism is introduced into the federated learning framework to effectively improve the performance of the model.

For data security protection, existing research has focused on data encryption and decryption techniques [18] and differential privacy techniques [19]. The former favors encryption of user data and protection of confidentiality during data transmission, while the latter favors protection of individual data and provision of privacy guarantees to enhance the data security of domain-wide power load forecasting systems. Differential privacy protects individual privacy while maintaining data availability by introducing noise into the data to obscure specific information about individual data, an approach that is effective against data analysis and inference and provides comprehensive privacy guarantees.

1.3. Research Motivation and Objectives

Although a great deal of research has been conducted in the existing literature on power load forecasting, as well as data protection security and federated learning, the existing techniques are still inadequate in addressing the challenges faced by new power systems, especially with the explosive growth of data in smart energy measurement systems and the complex issues of data quality and security. In addition, to address the need for data security and privacy protection in novel power systems, current research has not achieved sufficient integration in areas such as federated learning to ensure security and privacy protection during information exchange and model training.

To address the above problems, this paper is devoted to the study of power load forecasting and data security protection, as well as the fusion of the use of a federated learning system. Therefore, this paper proposes a federated learning system based on IHPO-WNN for whole-domain power load forecasting, which provides a more comprehensive and reliable solution for load forecasting and data security problems in power systems.

2. Power Load Forecasting Model Based on IHPO-WNN

In this chapter, a new novel prediction model combining improved hunter–prey optimizer (HPO) and wavelet neural network (WNN) is proposed, which has better prediction accuracy and robustness than other similar algorithms and can be used for prediction and analysis of electric power data such as local electric power loads in station areas. Firstly, improvement mechanisms such as Sine chaos mapping, dynamic boundaries, and parallel search mechanisms are mechanisms in the HPO algorithm to enhance its convergence performance and whole-process optimization capability. The above-improved algorithms are further combined with the wavelet neural network model to optimize the weights and wavelet factors of the WNN using the searching ability of the heuristic algorithm, hence enhancing the generalization ability and prediction performance of the model.

2.1. Wavelet Neural Network

WNN is developed based on the backpropagation neural network (BP neural network), which is a hybrid neural network model that combines the wavelet analysis method with the artificial neural network. It is characterized by replacing the transfer function in the nodes of the implicit layer with a wavelet basis function, which has certain advantages and features in processing signals and data. Figure 1 shows the topology of the three-layer wavelet neural network.

Assuming that there are n input layer nodes, p hidden layer nodes, and m output layer nodes, the input h_{ji} and output h_{jo} of one of the hidden layer nodes h_j can be represented as:

$$h_{ji} = \sum_{i=1}^{n} W_{ij} x_i, \tag{1}$$

$$h_{jo} = H\left(\frac{h_{ji} - b_j}{a_j}\right), \tag{2}$$

where a_j is the scaling factor of the basis function, b_j is the translation factor of the basis function, W_{ij} denotes the weight between the nodes of the input layer and the nodes of

the implicit layer, and $H(x)$ is the wavelet basis function, which is generally selected as the Morlet mother wavelet basis function, and is calculated as follows:

$$H(x) = \cos(1.75x)\exp(-\frac{x^2}{2}), \tag{3}$$

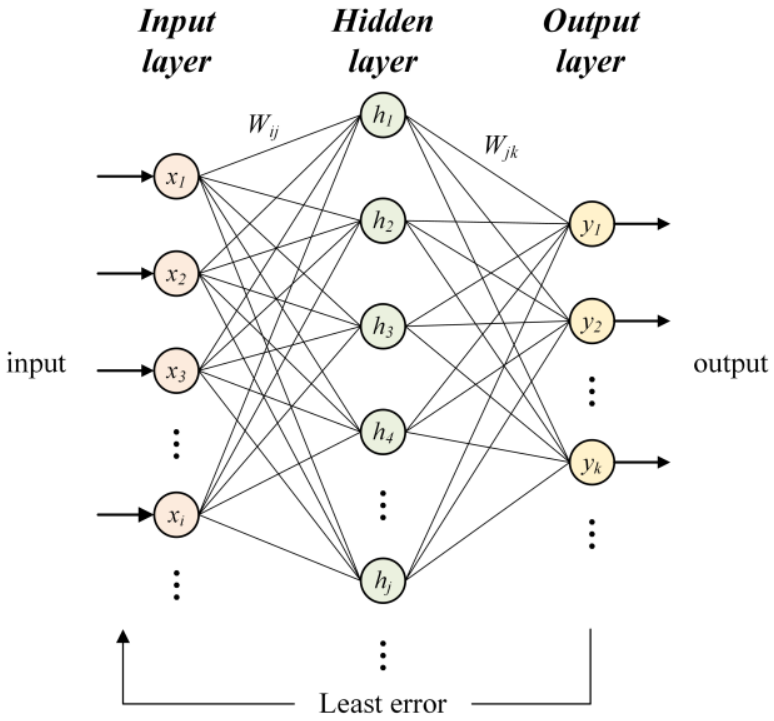


Figure 1. Topology of wavelet neural networks.

Thus, the output of the output layer node y_k of the network model can be obtained as in Equation (4), where W_{jk} is the weight between the implicit layer node and the output layer node.

$$y_k = \sum_{j=1}^{p} W_{jk} h_{jo}, \tag{4}$$

2.2. Hunter-Prey Optimizer

HPO is a new swarm intelligence optimization algorithm proposed by Iraj Naruei et al. in 2022 [20], which performs adaptive updating and optimization of the problem by simulating the behaviors of predators such as lions, leopards, and wolves, as well as prey such as deer and antelopes in nature. It has the characteristics of fast convergence and strong optimization-seeking ability.

The algorithm initializes the population by randomization, as shown in Equation (5):

$$X = rand(Np, dim) \times (ub - lb) + lb, \tag{5}$$

where Np denotes the population size, dim denotes the dimension of the problem variable, i.e., the sum of the weights between the layers of the WNN and the number of implicit layers stretching and leveling factors, and ub and lb denote the upper and lower bounds of the values taken, respectively.

HPO regulates the update amplitude by an adaptive parameter Z, which is calculated as shown in Equation (7):

$$P = R_1 < C;\ IDX = (P == 0), \tag{6}$$

$$Z = R_2 \otimes IDX + R_3 \otimes (\sim IDX), \tag{7}$$

where R_1 and R_3 are random vectors in [0, 1], respectively, R_2 is a random number in [0, 1], P is the index value of $R_1 < C$, and IDX is the index value of the vector R_1 that satisfies ($P == 0$). C is an iterative parameter for balancing global and local optimization seeking, which is computed as follows:

$$C = 1 - t(\frac{0.98}{T_iter}), \tag{8}$$

where t is the current number of iterations and T_iter is the maximum number of iterations.

HPO finds the optimal solution by iterating and updating the position of the population. The parameter β and a random number R_5 within [0, 1] are utilized to control the updating strategy and choose whether to update the population by hunter or prey, which is calculated as follows. Multiple benchmark test functions are utilized for simulation comparison, and the parameter β is set to 0.8 in this paper.

$$X(t+1) = \begin{cases} X(t) + 0.5\left[(2CZP_{pos} - X(t)) + 2(1-C)Z\mu - X(t)\right] & , R_5 < \beta \\ T_{pos} + CZ\cos(2\pi R_4) \times (T_{pos} - X(t)) & , else \end{cases}, \tag{9}$$

where $X(t + 1)$ is the position of the updated individual, $X(t)$ is the pre-update position, R_4 is a random number in [0, 1], and T_{pos} is the position of the globally optimal individual. Where μ denotes the average position of individuals in the population and P_{pos} denotes the position of the prey, the formulas are calculated as follows, respectively:

$$\mu = \frac{1}{Np}\sum_{i=1}^{Np} X_i, \tag{10}$$

$$P_{pos} = X_i | i \text{ is sorted } D_{euc}(kbest), \tag{11}$$

$$kbest = round(C \times Np), \tag{12}$$

$$D_{euc}(i) = \left(\sum_{j=1}^{dim} (X_{i,j} - \mu_j)^2\right)^{\frac{1}{2}}, \tag{13}$$

where X_i denotes the position of individual i in the population, $kbest$ is a decreasing control parameter, and $kbest = N$ at the beginning of the algorithm. $D_{euc}(i)$ denotes the distance between each individual and the average position, and Equation (11) denotes that the individuals in the population are sorted according to the position of the prey. $X_{i,j}$ denotes the position of individual i in the jth dimension, and μ_j is the average value of the position in the jth dimension.

2.3. Improved Hunter–Prey Optimizer

Considering the problems of poor initial population diversity in traditional HPO algorithms and degradation of convergence performance at the later stage of the algorithm, this paper proposes an improved HPO algorithm (IHPO) based on Sine chaotic mapping, dynamic cross-boundary, and parallel search mechanism. Firstly, an individual coding method is designed so that IHPO can finely optimize the parameters in WNN. Secondly, to improve the convergence speed of the algorithm, the Sine chaotic sequence is mapped to the solution space to generate the initial population of the algorithm. A parallel search mechanism based on dynamic boundaries is further designed to maximize the utilization of the optimization potential of each individual in the population. Specifically, the population is sorted according to the fitness value and divided into three sub-populations using

dynamic boundaries, and the sub-populations are each searched in parallel using different optimization strategies.

2.3.1. Coding and Decoding of Individuals

To integrate HPO with WNN, it is necessary to design the mapping relationship between the parameters of the neural network model and the individual coding of the heuristic algorithm. In this paper, we utilize the efficient optimization-seeking performance of HPO to optimize the weights between the layers of the WNN as well as the scaling and translation factors of the wavelet function. The design of the individual coding in the algorithm is shown in Figure 2:

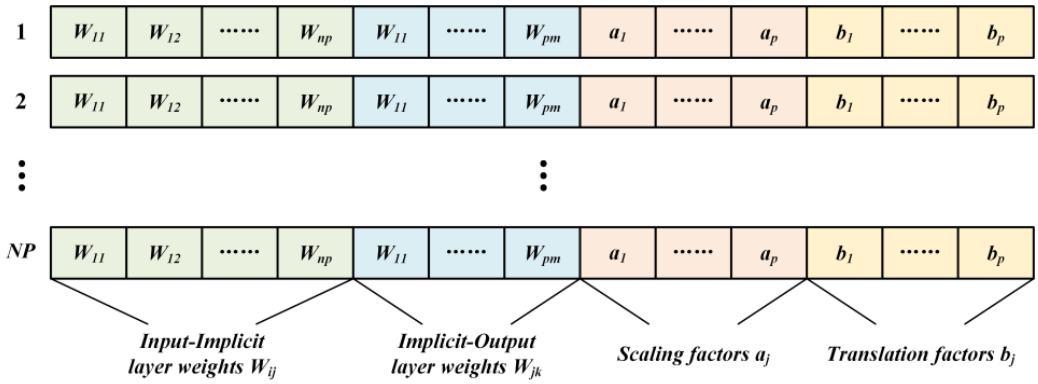


Figure 2. Forms of individual coding in algorithms.

In the IHPO optimization process, each optimization parameter is iterated as a whole in the form of Figure 2. While calculating the individual fitness value and training the WNN model, the above individual codes are reduced to specific parameters by decoding. By this method, each parameter in the WNN can be precisely adjusted, thus realizing the organic combination of the two types of algorithms.

2.3.2. Population Initialization Based on Sine Chaotic Mapping

Since the traditional HPO generates the initial population randomly, it is difficult to ensure the diversity of the population, and the uneven or too-dense distribution of individuals will affect the convergence speed and solution quality of the algorithm. The chaotic sequence has good traversability and arbitrariness, which can solve the problems of the above methods. Therefore, this paper introduces Sine chaotic mapping to initialize the population based on the traditional HPO, which can effectively improve the algorithm search efficiency and solution accuracy.

Sine mapping is one of the classical representatives of chaotic mapping, which has the advantages of simple structure, good chaotic properties, and fast generation speed compared with tent mapping, logistic mapping, and other chaotic mappings [21]. It can map the initial individuals uniformly on the whole solution space without repetition, thus ensuring the quality and diversity of the initial population. The specific generation formula of the Sine chaotic sequence is shown in Equation (14):

$$x_{i+1} = \rho \sin(\pi x_i),\ x \in [-1, 1], \tag{14}$$

where ρ is the chaos parameter; generally, take ρ = 1. At this time, the sequence will be in a completely chaotic state, as shown in Figure 3, and the values of the sequence will be uniformly distributed between [0, 1]. The generated chaotic sequence is mapped to the solution space of individuals as shown in Equation (15), and a high-quality initial population can be obtained:

$$X_i = lb + (1 + x_i) \times (ub - lb)/2 \tag{15}$$

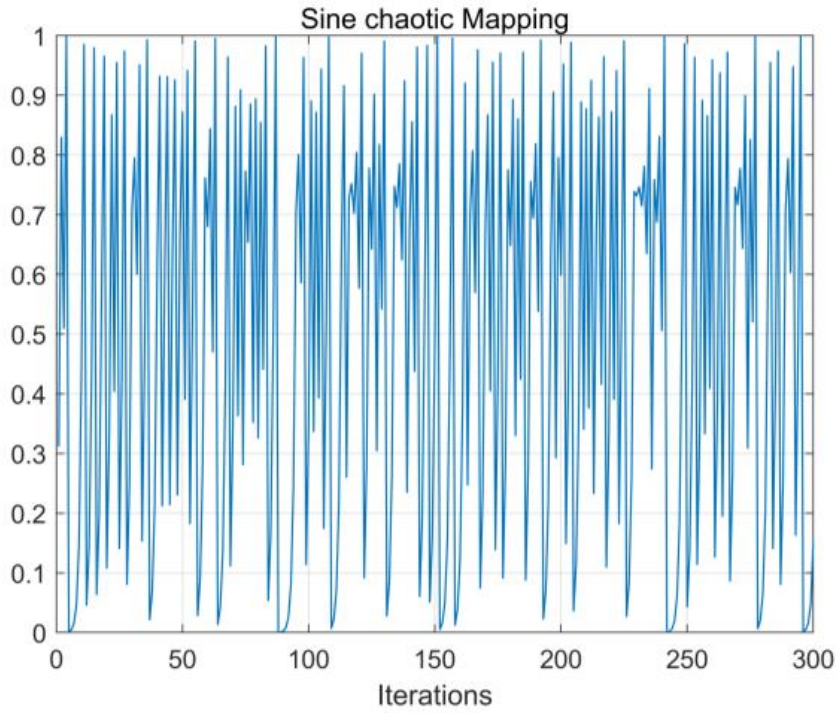


Figure 3. Sine chaotic sequence.

2.3.3. Parallel Search Mechanism Based on Dynamic Boundaries

To refine the division of the algorithmic population and maximize the use of each individual's search capability, this paper proposes a dynamic boundary mechanism for HPO. This mechanism dynamically stratifies the population and performs a parallel search between each stratum, which can effectively improve the diversity of the population. The hierarchical process is shown in Equation (16). Firstly, each individual in the population is sorted according to the fitness value from the largest to the smallest, and population X is divided into three sub-populations, elite population $X_{i,a}$, intermediate population $X_{i,b}$ and inferior population $X_{i,c}$ by the dynamic boundaries m and n, and the three sub-populations are updated in different ways at the same time, respectively.

$$\begin{cases} X_{i,a}, a = 1, 2, \cdots, m & Elite\ Guidance\ Update \\ X_{i,b}, b = m+1, m+2, \cdots, n & Original\ HPO\ Update \\ X_{i,c}, c = n+1, n+2, \cdots, N & Destructive\ Perturbation \end{cases} \tag{16}$$

Equation (17) is the updating formula of the dynamic boundary. Figure 4 shows the change curve of the dynamic cross-boundary when *T_iter* is taken as 1000 and *NP* is taken as 200, from which it can be seen that the number of the three sub-populations and the selection range show nonlinear changes with the iteration of the algorithm, in which the intermediate population size is always stable, while the elite population size dynamically decays and the inferior population size dynamically increases, thus realizing that the algorithm maintains a balance between global traversal and local optimal search.

Through this mechanism, the algorithm no longer adopts a single mechanism to update the individuals but adopts the optimal strategy according to the individual strengths and weaknesses, which is conducive to promoting the exchange of information between populations and enhancing the algorithm's global search capability.

$$\begin{cases} m = ceil\left[\frac{Np}{4} + Np(0.5 - \frac{t}{T_iter})^3\right] \\ n = ceil\left[\frac{3Np}{4} + Np(0.5 - \frac{t}{T_iter})^3\right] \end{cases} \tag{17}$$

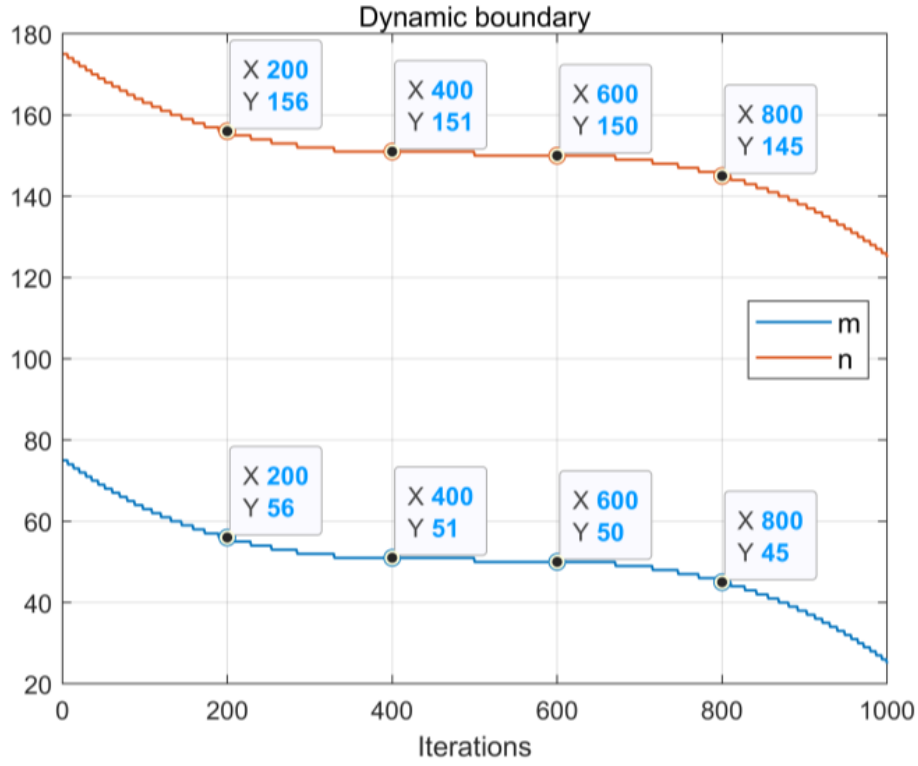


Figure 4. Dynamic cross-boundary change curves.

Following is a description of the specific updates after division into the three subpopulations.

(1) Intermediate population: traditional HPO updating approach

For the intermediate population $X_{i,b}$, the individuals' fitness is between better and worse, we chose not to exert too much intervention on its updating strategy, so the original HPO's individual updating formula is still used, as shown in Equation (9).

(2) Elite population: elite guidance strategy

For the elite population $X_{i,a}$, which has a high degree of adaptation in the population, we use the elite bootstrap strategy to update it and try to play the role of the leader of the globally optimal solution without destroying the advantageous individual positions and exploiting the advantageous positions in the solution space more fully. The specific way is to introduce individual extremes based on the HPO updating formula, Equation (9), as shown in Equation (18), thus accelerating the convergence speed of the head population.

$$X(t+1) = \begin{cases} X(t) + 0.5\left[(2CZP_{pos} - X(t)) + 2(1-C)Z\mu - X(t)\right] & , R_5 < \beta \& R_6 < 0.5 \\ T_{pos} + 0.5\left[(2CZP_{pos} - T_{pos}) + 2(1-C)Z\mu - T_{pos}\right] & , R_5 < \beta \& R_6 \geq 0.5 \\ T_{pos} + CZ\cos(2\pi R_4) \times (T_{pos} - X(t)) & , else \end{cases}, \quad (18)$$

(3) Disadvantaged populations: destructive perturbation strategy

For the disadvantaged population $X_{i,c}$, its individuals can hardly play an effective role in the algorithm's optimization, so it is necessary to adopt a more "radical" update strategy. In this paper, we use a combination of backward learning and Gaussian variation to update the population with destructive perturbations. Reverse learning [22] means generating a reverse solution randomly in the solution space according to the position of the original solution, whose randomness is controlled by the random number k. The Gaussian variation strategy [23] can effectively enhance the local and global optimization-seeking ability of the algorithm and has been widely used in various meta-heuristic algorithms for improvement

strategies. The destructive perturbation update based on backward learning and Gaussian variation is shown in Equation (19):

$$X(t+1) = \begin{cases} k(ub+lb) - X(t) & R_7 < 0.5 \\ X(t) + Gaussian(\mu, \lambda^2) & R_7 \geq 0.5 \end{cases}, \quad (19)$$

where k is set as a random number between [0, 1], μ denotes the mean, and λ^2 denotes the variance. Considering the range of values of weights and wavelet factors, this paper sets $\mu = 0$ and $\lambda = 1$.

2.4. IHPO-WNN

In summary, the overall flowchart of a power load forecasting model based on an improved HPO-optimized wavelet neural network (IHPO-WNN) proposed in this paper is shown in Figure 5.

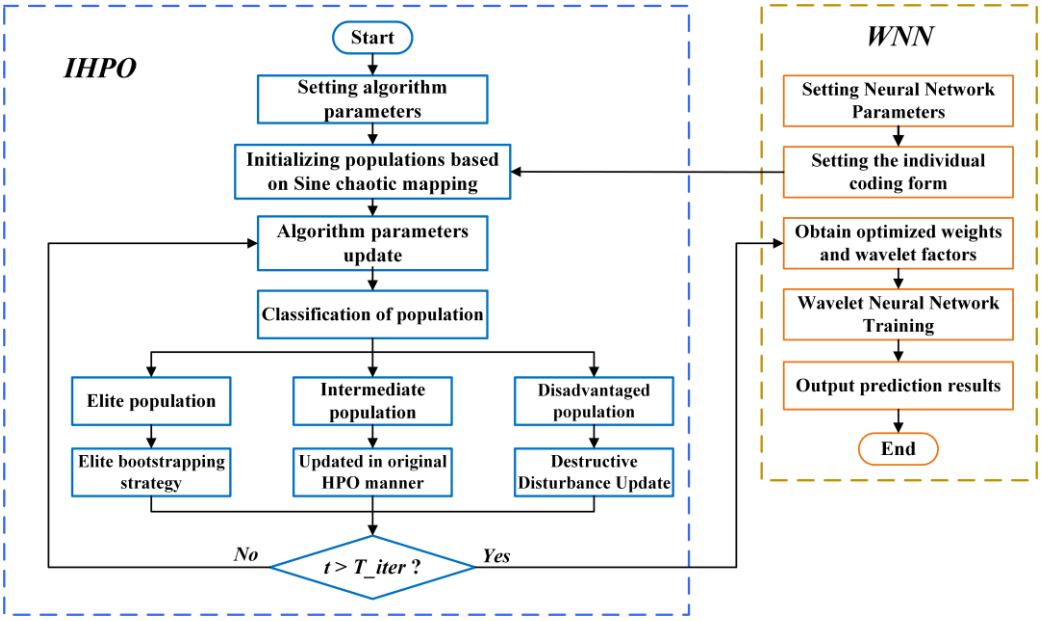


Figure 5. IHPO-WNN Overall Flowchart.

3. An IHPO-WNN-Based Federated Learning Architecture for Power Data Prediction

In this chapter, an IHPO-WNN-based federated learning architecture for power load data prediction is proposed to realize the completion of the prediction and analysis of power metering data in the whole domain without direct data exchange in each station. The above IHPO-WNN prediction model is used as the base model to improve the prediction performance of the model. Meanwhile, to ensure the security of the communication between the metering master and each station, the uploaded signals are encrypted by the localized differential privacy method, which can effectively resist the threat of potential network attacks.

3.1. Federated Learning Based on IHPO-WNN

Federated learning is a distributed machine learning framework consisting of a participant, a central aggregator, and a communication network, which can be a good solution to the problem of information silos in new power systems [24]. It is characterized by the fact that the participants do not directly upload raw data but only model training parameters, which can significantly reduce the risk of data leakage. In this paper, the above IHPO-WNN is used as the prediction base model of federated learning, the stations are treated as local participating users, and the metering master is treated as the data aggregator to realize

the accurate prediction of power metering data in a wide area, and the specific process is as follows:

(1) Local model initialization

Each station generates local optimization parameters ω_k based on local data through IHPO-WNN and uploads the parameters to the metering master station under its jurisdiction for aggregation.

(2) Metering master center aggregation

After receiving the model parameters uploaded by each station participating in the federated learning, the metrology master performs the first parameter fusion update through the federated averaging algorithm to obtain the first global model parameters, which are re-distributed to each station for local training, as shown in Equation (20):

$$\omega = \sum_{k=1}^{N} \frac{n_k}{n} \omega_k, \tag{20}$$

where n is the total data volume of all stations, n_k is the data volume of station k, N is the number of stations participating in federated learning, and ω_k is the local parameters uploaded by each station.

(3) Local training process

After each station receives the first global parameter from the metering master, based on the WNN, it performs local updating based on this parameter by the gradient descent method and uploads the updated local parameters again. For the ith parameter of the local station k, the update is performed by Equation (21):

$$\omega_k^i \leftarrow \omega_k^i - \eta \nabla f_k(\omega_k^i), \tag{21}$$

$$\nabla f_k(\omega_k^i) = \frac{\partial f_k(\omega_k^i)}{\partial \omega_k^i}, \tag{22}$$

where η denotes the learning rate and $f_k(\cdot)$ is the loss function of the model.

(4) Aggregate again to get the final model

Each station uploads the locally trained parameters, and the metering master carries out the second aggregation according to Equation (20) to obtain the final global prediction model parameters, which are decentralized to each station to verify the feasibility.

3.2. Channel Encryption Mechanism Based on Localized Differential Privacy

Although the above federated learning mechanism, to a certain extent, has realized the privacy protection of the data of each station, its privacy protection presupposes that the metering master is trustworthy as well as that the information will not be intercepted by the attacker during the transmission process. If the model parameters are intercepted by an attacker during the uploading process from the station to the metering master, it is easy to obtain the original private data through methods such as query differences or inference attacks.

The differential privacy technique, on the other hand, can well solve the above problem, which is centered on data distortion, and can make each user's private data not vulnerable to leakage due to the attack on other data in the system. Therefore, this paper introduces local differential privacy (LDP) to encrypt the upload parameters of each station in Laplace, so that the parameters satisfy the ε-differential privacy, to further the privacy enhancement of the above-federated learning framework. Among the features of localized differential privacy is that local stations encrypt the parameters before uploading, which solves the problem of needing to trust the aggregator in centralized differential privacy.

The mathematical definition of ε-differential privacy [25] is as follows: given any two neighboring datasets, Data and Data′, an algorithm M is said to satisfy ε-differential privacy if the algorithm M satisfies Equation (23):

$$\frac{\Pr[M(Data) = S]}{\Pr[M(Data') = S]} \leq e^{\varepsilon}, \tag{23}$$

where *Pr[*]* denotes the probability and ε denotes the privacy budget. Adding noise is one of the main methods to realize differential privacy and the Laplace mechanism is a noise-adding mechanism to satisfy ε-differential privacy. The probability density of this noise is shown in Equation (24):

$$f(x\Big|\mu,\sigma) = \frac{1}{2\sigma}\exp\left(-\frac{|x-\mu|}{\sigma}\right), \tag{24}$$

where μ is the mean of the Laplace distribution and σ is a scale parameter representing the size of the added noise and the degree of privacy encryption. The Laplace encryption process is shown in Equation (25), which is realized by adding random noise to the real output result to satisfy the ε-differential privacy in the case that the attacker is unable to reverse the original data even after intercepting the uploaded information.

$$Y(X) = f(X) + Lap(\mu,\sigma), \tag{25}$$

where $f(X)$ is the original output, $Lap(\mu,\sigma)$ is the added Laplace noise, and $Y(X)$ is the parameter that is passed to the central aggregator. If the privacy budget is set smaller, i.e., the parameter σ is set smaller, the data is encrypted to a higher degree but in contrast, more noise is introduced, and the accuracy of the model decreases. Taking the standard normal distribution function as the original data, as shown in Figure 6, a comparison plot of the effect of Laplace noise on the data with different privacy budgets added is shown.

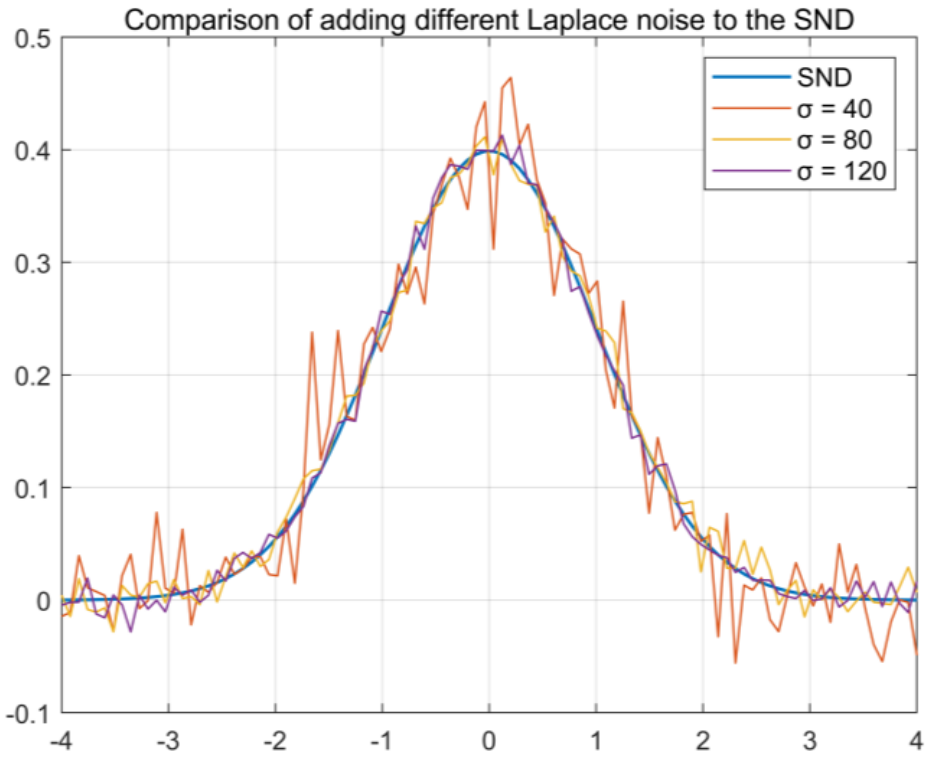


Figure 6. Impact of adding noise with different privacy budgets on data.

3.3. An IHPO-WNN-Based Federated Learning System for Area-Wide Power Load Forecasting

Based on the above, the whole-domain prediction system for power metering data based on IHPO-WNN and federated learning architecture is obtained, as shown in Figure 7, which can be used for whole-domain prediction and analysis of all kinds of distributed confidential data.

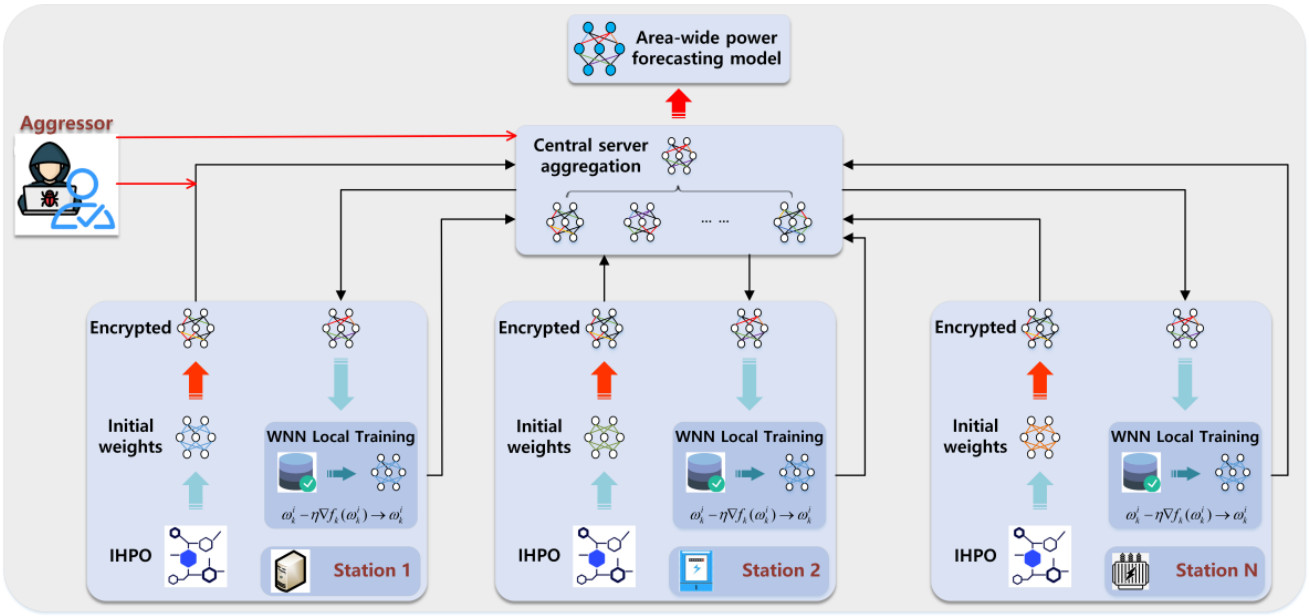


Figure 7. A federal learning system for area-wide power load forecasting.

4. Experimental Results and Analysis

4.1. Description of the Dataset

The dataset used in this paper comes from the smart energy measurement master station of a power grid company's metering center. The power load data from seven different stations from 0:00 on 1 May 2023 to 23:45 on 30 June 2023 are selected for this paper; the data types are all low-voltage civil loads and the data sources and statistical information are shown in Table 1. The data collection interval is 15 min; that is, there are seven groups of power load data—each group has 5856 pieces of data volume, the total data volume is 40,992 pieces of data, and the data is presented as shown in Figure 8. From the figure, the power load of the distribution transformer shows obvious nonlinearity and a certain periodical pattern, with the peak of power consumption occurring at noon time every day and the trough of power consumption at midnight, which is consistent with the actual situation. Meanwhile, the load data of each station has similar distribution characteristics, which makes the model in this paper have better generalization ability when facing the data of different stations.

Table 1. Sources of data sets and statistical analysis.

Station 1 **Unique Station Identification XXXX7289**			**Station 2** **Unique Station Identification XXXX7324**		
Average Load	Max Load	Min Load	Average Load	Max Load	Min Load
0.6240 kW	1.2616 kW	0.1587 kW	1.6218 kW	2.8466 kW	0.5214 kW
Station 3 **Unique station identification XXXX7328**			**Station 4** **Unique station identification XXXX7355**		
Average Load	Max Load	Min Load	Average Load	Max Load	Min Load
1.2828 kW	2.9747 kW	0.4067 kW	1.0616 kW	1.8219 kW	0.4221 kW

Table 1. *Cont.*

Station 5 Unique station identification XXXX7552			**Station 6 Unique station identification XXXX7908**		
Average Load	Max Load	Min Load	Average Load	Max Load	Min Load
0.4152 kW	1.3594 kW	0.1809 kW	1.1681 kW	2.3307 kW	0.3662 kW
Station 7 Unique station identification XXXX7914					
Average Load		Max Load		Min Load	
0.6781 kW		2.0028 kW		0.1197 kW	

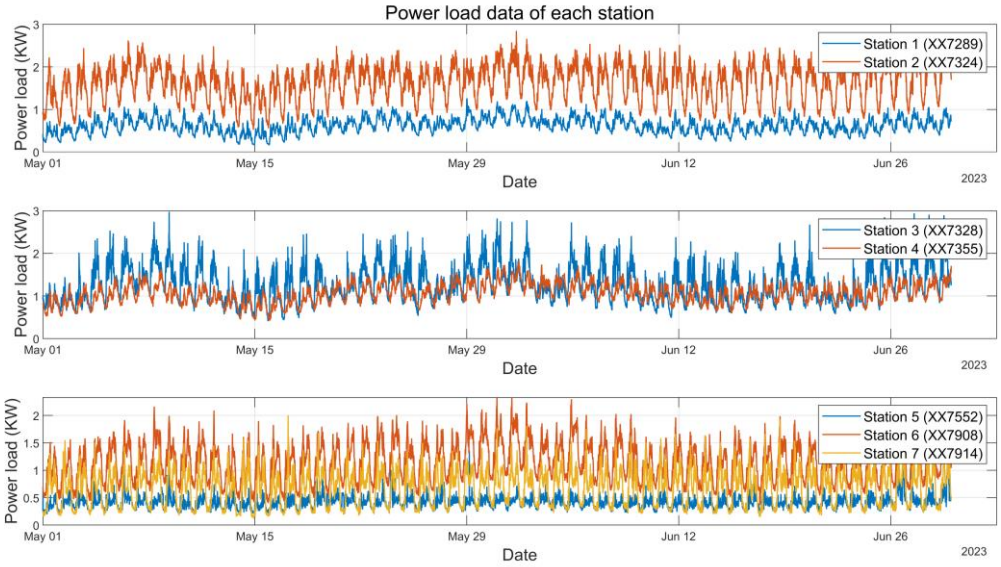


Figure 8. Power load data from seven stations.

4.2. Forecast Evaluation Indicators

In this paper, Root Mean Square Error (*RMSE*), Mean Absolute Percentage Error (*MAPE*), and coefficient of determination R^2 are used to evaluate the performance of the forecasting model, thereby measuring the accuracy of the electricity load forecasting. Among them, *RMSE* measures the degree of difference between the model's predicted values and the actual observed values, which can reflect the overall prediction accuracy of the model; *MAPE* is used to evaluate the relative error of the model, which can measure the relative accuracy of the model. R^2 measures the degree of the model's explanation for the total variation, which reflects the degree of linear relationship between the model's predicted values and the actual values. The formulas for the three evaluation metrics are shown in Equations (26)–(28).

$$RMSE = \sqrt{\frac{1}{M}\sum_{i=1}^{M}(\hat{y}_i - y_i)^2}, \tag{26}$$

$$MAPE = \frac{100\%}{M}\sum_{i=1}^{M}\left|\frac{\hat{y}_i - y_i}{y_i}\right|, \tag{27}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{M}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{M}(y_i - y_{ave})^2}, \tag{28}$$

where M is the total number of samples, $\hat{y}_i$ is the load prediction value, y_i is the actual load value, and y_{ave} is the average value of the actual data. The closer the values of *RMSE* and *MAPE* are to 0, and the closer the value of R^2 is to 1, the better the prediction performance of the model is represented.

4.3. IHPO-WNN Performance Testing

Before testing the performance of the federated learning system, this paper first conducts a performance test on the prediction performance of the designed IHPO-WNN. The IHPO-WNN is applied to the data of Stations 1 and 2 in the described dataset for one-day and one-week load forecasting, respectively, and is compared with a series of cutting-edge forecasting models such as the unimproved HPO-optimized wavelet neural network model (HPO-WNN), the WNN, the Genetic Algorithm-optimized BP neural network model (GA-BP), the Long and Short-Term Memory recurrent neural network (LSTM), and the Generalized Regression Neural Network (GRNN), and a series of cutting-edge prediction models are compared side-by-side.

The software environment for the simulation experiments in this paper is MATLAB R2022a, the computer operating system used is Windows 11, the memory is 16 GB RAM, the CPU is AMD 5800U, and the graphics card is NVIDIA GeForce RTX 3050 Laptop GPU. The parameters of each algorithm are set as follows: a learning rate of 0.01 for the scaling factor a_j, a learning rate of 0.001 for the translation factor b_j, and several nodes in the implicit layer of 32 in the WNN, a learning rate of 0.01 and several nodes in the implicit layer of 32 in the GA-BP, and an initial learning rate of 0.005 in the LSTM, with two implicit layers and the number of nodes in each layer of 20.

To ensure the fairness of the simulation experiment, the number of training times of each neural network model is set to 100; when meta-heuristic algorithms are involved, the maximum number of iterations of the algorithms is set to 50, and the size of the populations is set to 50. As shown in Figures 9 and 10, the daily and weekly prediction results of the algorithms are compared with the real value of the electric power loads, and Tables 2–5 are the evaluation indexes computed from the prediction results of the groups, respectively.

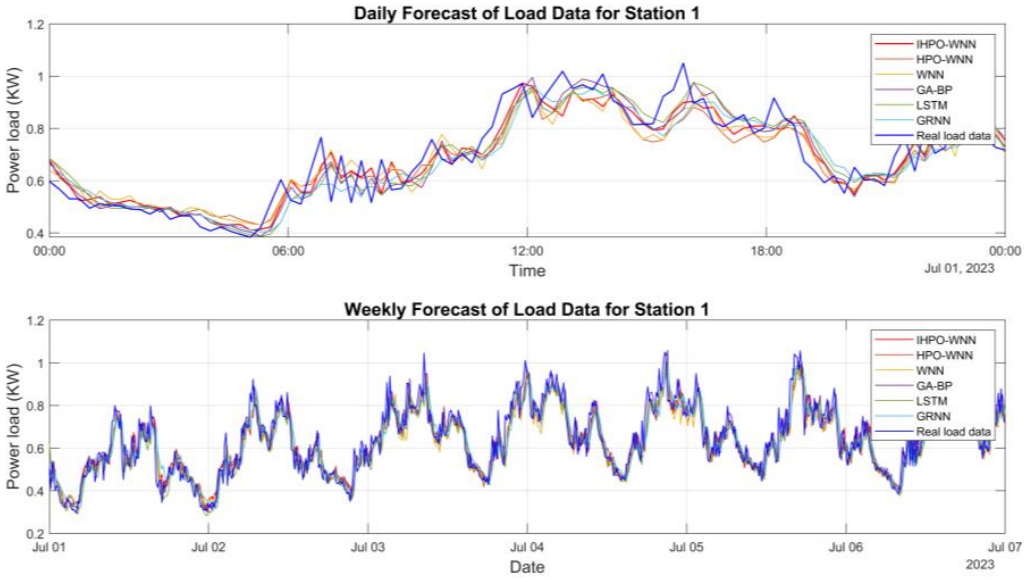


Figure 9. Daily and weekly forecast results for Station 1.

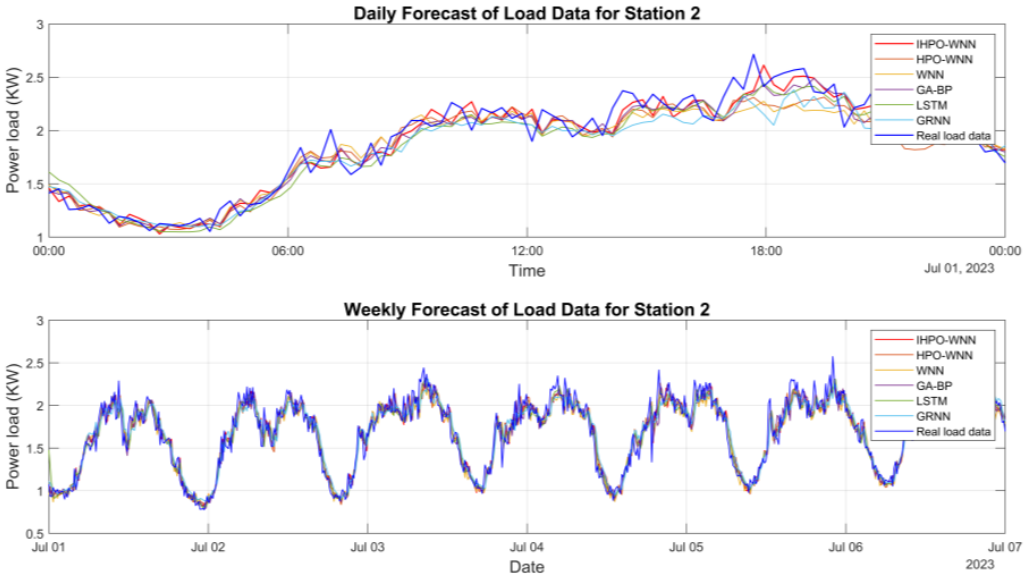


Figure 10. Daily and weekly forecast results for Station 2.

Table 2. Evaluation indicators for daily forecasts of Station 1.

	IHPO-WNN	HPO-WNN	WNN	GA-BP	LSTM	GRNN
RMSE	0.071059	0.074338	0.07892	0.07488	0.074069	0.078006
MAPE	8.3034%	8.7870%	9.4693%	8.5060%	8.5247%	8.5569%
R^2	0.8355	0.81997	0.79709	0.81730	0.82127	0.80177

Table 3. Evaluation indicators for weekly forecasts of Station 1.

	IHPO-WNN	HPO-WNN	WNN	GA-BP	LSTM	GRNN
RMSE	0.068131	0.068972	0.07287	0.68969	0.069701	0.073649
MAPE	8.2140%	8.2149%	8.4293%	8.3331%	7.9723%	8.8684%
R^2	0.83038	0.82617	0.80597	0.82619	0.82248	0.80180

Table 4. Evaluation indicators for daily forecasts of Station 2.

	IHPO-WNN	HPO-WNN	WNN	GA-BP	LSTM	GRNN
RMSE	0.12965	0.14656	0.15733	0.13444	0.14159	0.16062
MAPE	5.3901%	5.8837%	6.0895%	5.4373%	5.9587%	5.9088%
R^2	0.91393	0.89002	0.87326	0.90745	0.89734	0.86789

Table 5. Evaluation indicators for weekly forecasts of Station 2.

	IHPO-WNN	HPO-WNN	WNN	GA-BP	LSTM	GRNN
RMSE	0.13750	0.13996	0.14567	0.13941	0.13277	0.14335
MAPE	6.2056%	6.3356%	6.5528%	6.3054%	5.9875%	6.5828%
R^2	0.89669	0.89296	0.88404	0.89381	0.90368	0.88772

Based on the data in Tables 2–5, it can be clearly observed that the IHPO-WNN model significantly outperforms other similar algorithms in three metrics, namely, *RMSR*, *MAPE*, and R^2, on different datasets, regardless of whether they are forecasting electricity loads for a day or a week in the future. These results strongly validate the superior applicability of the IHPO-WNN model in the field of electricity load forecasting. Compared with the

unimproved HPO-WNN model, as well as using only the WNN model, the results also fully demonstrate the effectiveness of the improvement strategy proposed in this paper.

4.4. Simulation Testing of a Domain-Wide Load Factor Prediction Model

In this section, the performance of the federated learning system is tested to verify its feasibility, and the impact of introducing localized differential privacy on the prediction performance of the system is also analyzed. The seven stations in the dataset are used as the seven federated learning participants, and the metering master station is used as the central aggregator, assuming that the stations and the master station communicate with each other through a wireless network, and the simulation experiments are conducted according to the steps in Section 3.1 with the same algorithm parameters as above, and the privacy budget parameter σ set to 100.

Simulations are performed for the two cases of differential privacy encryption and no encryption, respectively, and the resulting global model is decentralized to each station for prediction. Some of the prediction results are shown in Figure 11, a comparison of the various metrics of the prediction results for the two cases is shown in Table 6, and Figure 12 shows a comparison of the absolute error curves of the model training carried out at each station in the two cases.

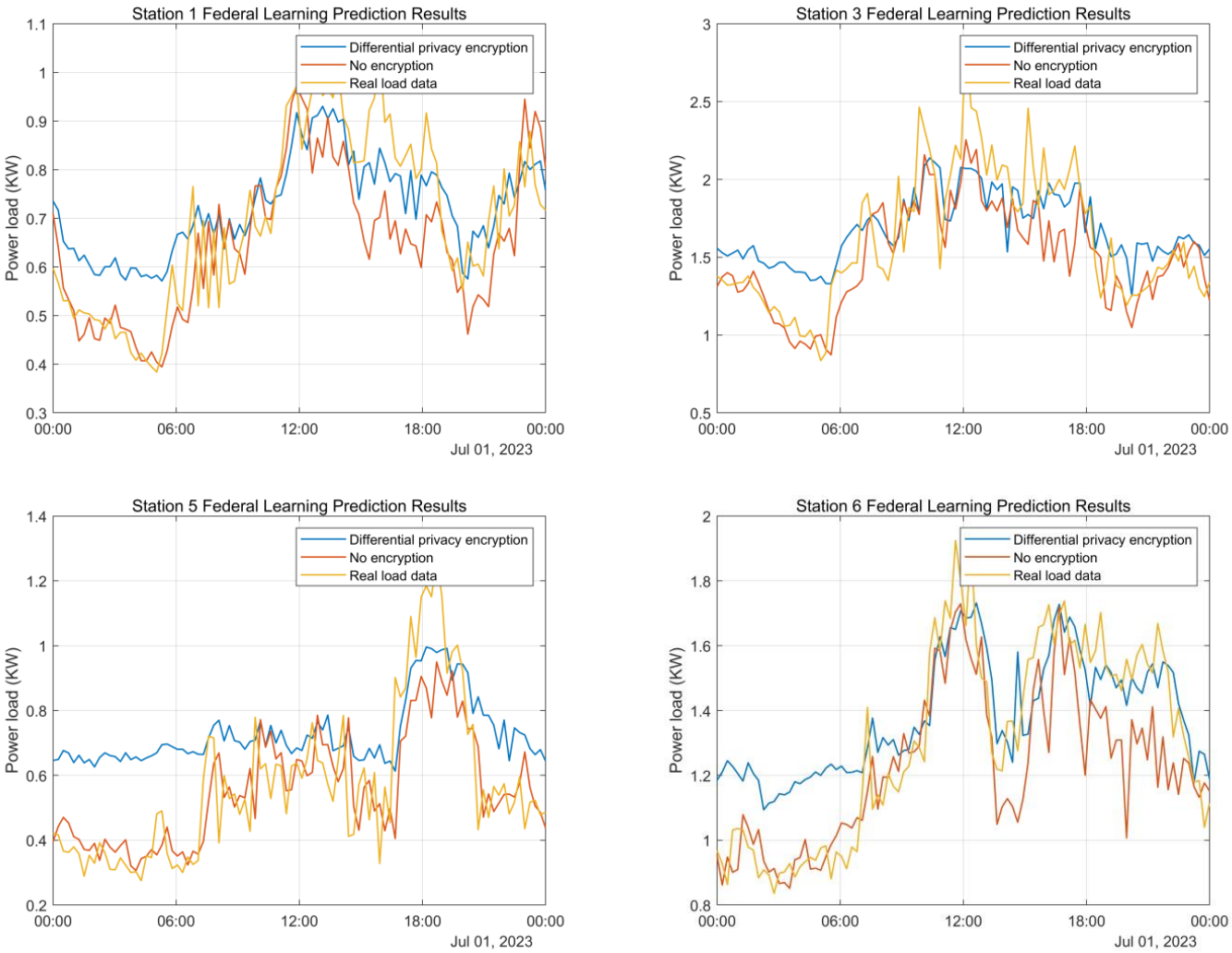


Figure 11. Federal learning partial station forecast results.

Table 6. Federal learning system predictors.

Station Number	Differential Privacy			Unencrypted		
	RMSE	*MAPE*	R^2	*RMSE*	*MAPE*	R^2
Station 1	0.10440	15.0973%	0.64492	0.11896	12.6128%	0.53900
Station 2	0.31743	13.7480%	0.48405	0.37544	14.7143%	0.27826
Station 3	0.28284	15.9380%	0.55777	0.26832	11.3850%	0.60200
Station 4	0.13668	8.6843%	0.60064	0.20990	12.5242%	0.05815
Station 5	0.22885	47.0206%	0.03619	0.14092	18.0187%	0.63453
Station 6	0.17789	13.3376%	0.64627	0.17994	9.7863%	0.63806
Station 7	0.36251	71.3801%	0.01856	0.15480	16.5710%	0.82104

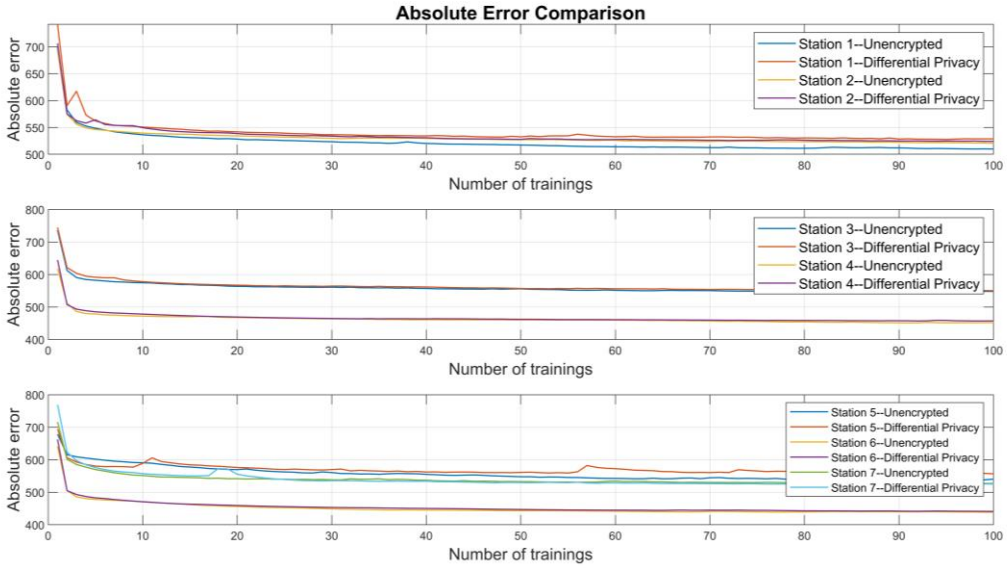


Figure 12. Comparison of absolute error curves before and after encryption.

From Table 6, when the global prediction model generated by the federated learning system is used for load forecasting in each station area, its accuracy slightly decreases relative to the direct local forecasting, and all the evaluation indexes are reduced. However, by and large, it can still achieve very good prediction results, while federated learning brings irreplaceable advantages for data privacy and global applicability.

In addition, from the above simulation results, compared with the unencrypted case, the encryption using localized differential privacy can greatly enhance the security of the data, although a small part of the prediction accuracy will be sacrificed. It is worth emphasizing that the data used in the above simulation experiments are all actual civil loads of low-voltage stations, and there is a certain similarity in the distribution pattern among the data of different stations, so the model in this paper shows good generalization ability among different stations. When this system is applied in practice, the accuracy of the prediction may fluctuate significantly if there is too much disparity in the characteristics of the data from different stations, e.g., covering heavy industrial loads, agricultural loads, and residential loads at the same time, resulting in inconsistency in the distribution among the data. Improvements such as personalized federal learning techniques can be considered to further improve the robustness of the model. Meanwhile, personalized parameter adjustments based on the degree of privacy of the data in each station area and the demand for prediction accuracy can achieve a balance between privacy and accuracy.

4.5. Sensitivity Analysis

In the above experiments, the privacy budget parameter σ is an important parameter that affects the prediction accuracy and privacy preservation, so it is necessary to analyze its sensitivity. Taking σ = 100 in Section 4.4 as the baseline, and fine-tuning it in steps of 20, four sets of full-domain simulation tests are conducted, respectively, and the results of the prediction metrics are shown in Table 7.

Table 7. Sensitivity analysis results.

Station Number	σ = 60			σ = 80		
	RMSE	***MAPE***	R^2	***RMSE***	***MAPE***	R^2
Station 1	0.11746	14.5860%	0.55053	0.13082	17.0072%	0.44249
Station 2	0.29947	13.5867%	0.54078	0.36354	15.9714%	0.32329
Station 3	0.31446	15.6211%	0.45336	0.30796	16.5139%	0.47572
Station 4	0.16259	11.2651%	0.43490	0.17466	10.8942%	0.34787
Station 5	0.18931	32.3783%	0.34048	0.18644	34.9640%	0.36027
Station 6	0.20232	12.9618%	0.54241	0.18584	13.5295%	0.61394
Station 7	0.39531	67.8287%	0.16707	0.25223	45.7235%	0.52485
Station Number	σ = 120			σ = 140		
	RMSE	***MAPE***	R^2	***RMSE***	***MAPE***	R^2
Station 1	0.12746	18.3574%	0.47070	0.10420	14.8352%	0.64627
Station 2	0.33395	15.8748%	0.42895	0.21694	10.9808%	0.75903
Station 3	0.33014	19.7675%	0.39747	0.27449	15.8748%	0.58350
Station 4	0.15127	10.3784%	0.51085	0.11061	7.42880%	0.73843
Station 5	0.24279	29.4584%	0.08484	0.23452	16.0809%	0.52153
Station 6	0.21546	16.0744%	0.48104	0.36633	12.4916%	0.68007
Station 7	0.28079	47.7928%	0.41117	0.15480	37.8438%	0.52387

From the table, as the privacy budget parameter σ increases, the prediction accuracy improves to a certain extent, while in contrast, the privacy preservation decreases, which is consistent with the theoretical results. However, within a certain range, the variation of σ does not significantly affect the trade-off between prediction accuracy and privacy preservation. This finding further emphasizes its robustness as a key parameter and provides reliable theoretical support for its application in practice.

5. Conclusions

In this paper, we propose a localized differential privacy-based federated learning architecture for load forecasting, which effectively solves the problems of low data prediction efficiency, insufficient data security, and "data islands" between different stations in the current new power system. By combining the improved HPO algorithm with WNN, accurate prediction of power load data is achieved, and the experiments on different data sets show high prediction accuracy and generalization ability. Meanwhile, a localized differential privacy-based load prediction federated learning architecture for global prediction of power loads is proposed to protect sensitive data and resist the threat of cyber-attacks, which can still maintain high prediction accuracy while safeguarding data privacy, reflecting the balance between privacy protection and prediction accuracy. In addition, the data used in this paper are all from the actual dataset of a metering center's smart energy measurement master, which ensures the validity and authenticity of the described work.

The work in this paper focuses on the combination of optimization algorithms and neural networks as well as information encryption technology, which provides useful guidance and ideas for the intelligent development of power systems. However, there are still shortcomings in this study, such as not considering the data characteristics and distribution differences of different stations, which led to a decrease in the prediction performance of the federated learning system on some stations, and not conducting a more

in-depth study on the applicability of some specific data scenarios. Future work can further explore the parameter tuning methods applicable to different scenarios, apply the method to other related field scenarios, and also consider extending the method of this paper to larger-scale real power systems to further enhance its performance and practicality.

Author Contributions: Conceptualization, B.S. and N.P.; methodology, B.S. and X.Z.; software, X.Z.; validation, B.S. and P.L.; formal analysis, X.Z.; resources, B.S. and N.P.; data curation, W.M.; writing—original draft preparation, B.S. and X.Z.; writing—review and editing, X.Z.; visualization, X.Z. and W.M.; supervision, N.P.; project administration, P.L.; funding acquisition, P.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the science and technology project of China Southern Power Grid Co., Ltd. under Grants YNKJXM20220111.

Data Availability Statement: Not applicable.

Code Availability Statement: The source code is available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Jiang, Y.; Gao, T.; Dai, Y.; Si, R.; Hao, J.; Zhang, J.; Gao, D.W. Very short-term residential load forecasting based on deep-autoformer. *Appl. Energy* **2022**, *328*, 120120. [CrossRef]
2. Cai, Z.; Dai, S.; Ding, Q.; Zhang, J.; Xu, D.; Li, Y. Gray wolf optimization-based wind power load mid-long term forecasting algorithm. *Comput. Electr. Eng.* **2023**, *109*, 108769. [CrossRef]
3. Jahani, A.; Zare, K.; Khanli, L.M. Short-term load forecasting for microgrid energy management system using hybrid SPM-LSTM. *Sustain. Cities Soc.* **2023**, *98*, 104775. [CrossRef]
4. Xia, Y.; Wang, J.; Zhang, Z.; Wei, D.; Yin, L. Short-term PV power forecasting based on time series expansion and high-order fuzzy cognitive maps. *Appl. Soft Comput.* **2023**, *135*, 110037. [CrossRef]
5. Behmiri, N.B.; Fezzi, C.; Ravazzolo, F. Incorporating air temperature into mid-term electricity load forecasting models using time-series regressions and neural networks. *Energy* **2023**, *278*, 127831. [CrossRef]
6. Atef, S.; Nakata, K.; Eltawil, A.B. A deep bi-directional long-short term memory neural network-based methodology to enhance short-term electricity load forecasting for residential applications. *Comput. Ind. Eng.* **2022**, *170*, 108364. [CrossRef]
7. Bu, X.; Wu, Q.; Zhou, B.; Li, C. Hybrid short-term load forecasting using CGAN with CNN and semi-supervised regression. *Appl. Energy* **2023**, *338*, 120920. [CrossRef]
8. Sitapure, N.; Kwon, J.S.-I. CrystalGPT: Enhancing system-to-system transferability in crystallization prediction and control using time-series-transformers. *Comput. Chem. Eng.* **2023**, *177*, 108339. [CrossRef]
9. Sitapure, N.; Kwon, J.S.-I. Exploring the potential of time-series transformers for process modeling and control in chemical systems: An inevitable paradigm shift? *Chem. Eng. Res. Des.* **2023**, *194*, 461–477. [CrossRef]
10. Zhang, Y.; Kong, L. Photovoltaic power prediction based on hybrid modeling of neural network and stochastic differential equation. *ISA Trans.* **2022**, *128*, 181–206. [CrossRef]
11. Xian, H.; Che, J. Multi-space collaboration framework based optimal model selection for power load forecasting. *Appl. Energy* **2022**, *314*, 118937. [CrossRef]
12. Lee, D.; Jayaraman, A.; Kwon, J.S. Development of a hybrid model for a partially known intracellular signaling pathway through correction term estimation and neural network modeling. *PLoS Comput. Biol.* **2020**, *16*, e1008472. [CrossRef] [PubMed]
13. Wu, M.; Zhong, Y.; Wu, J.; Wang, Y.; Wang, L. State of health estimation of the lithium-ion power battery based on the principal component analysis-particle swarm optimization-back propagation neural network. *Energy* **2023**, *283*, 129061. [CrossRef]
14. Zhang, H.; Xue, J.; Wang, Q.; Li, Y. A security optimization scheme for data security transmission in UAV-assisted edge networks based on federal learning. *Ad Hoc Netw.* **2023**, *150*, 103277. [CrossRef]
15. Chandiramani, K.; Garg, D.; Maheswari, N. Performance Analysis of Distributed and Federated Learning Models on Private Data. *Procedia Comput. Sci.* **2019**, *165*, 349–355. [CrossRef]
16. Guendouzi, B.S.; Ouchani, S.; EL Assaad, H.; EL Zaher, M. A systematic review of federated learning: Challenges, aggregation methods, and development tools. *J. Netw. Comput. Appl.* 2023; *in press*. [CrossRef]
17. Li, C.; Song, M.; Luo, Y. Federated learning based on Stackelberg game in unmanned-aerial-vehicle-enabled mobile edge computing. *Expert Syst. Appl.* **2024**, *235*, 121023. [CrossRef]
18. Urooj, S.; Lata, S.; Ahmad, S.; Mehfuz, S.; Kalathil, S. Cryptographic Data Security for Reliable Wireless Sensor Network. *Alex. Eng. J.* **2023**, *72*, 37–50. [CrossRef]
19. Errounda, F.Z.; Liu, Y. Adaptive differential privacy in vertical federated learning for mobility forecasting. *Future Gener. Comput. Syst.* **2023**, *149*, 531–546. [CrossRef]

20. Naruei, I.; Keynia, F.; Molahosseini, A.S. Hunter–prey optimization: Algorithm and applications. *Soft Comput.* **2022**, *26*, 1279–1314. [CrossRef]
21. Sun, L.; Li, M.; Xu, J. Binary Harris Hawk optimization and its feature selection algorithm. *Comput. Sci.* **2023**, *50*, 277–291. [CrossRef]
22. Mohapatra, S.; Mohapatra, P. Fast random opposition-based learning Golden Jackal Optimization algorithm. *Knowl.-Based Syst.* **2023**, *275*, 110679. [CrossRef]
23. Wei, C.; Wei, X.; Huang, H. Pigeon flocking algorithm based on chaotic initialization and Gaussian variation. *Comput. Eng. Des.* **2023**, *44*, 1112–1121. [CrossRef]
24. Fu, S.; Zhao, X.; Yang, C. Data heterogeneous federated learning algorithm for industrial entity extraction. *Displays* **2023**, *80*, 102504. [CrossRef]
25. Yang, M.; Cheng, H.; Chen, F.; Liu, X.; Wang, M.; Li, X. Model poisoning attack in differential privacy-based federated learning. *Inf. Sci.* **2023**, *630*, 158–172. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.