# Design of Federated Learning and Data Sharing System for Environmental Information System in Multi-Campus Environment

**Zhenjing Lin[1], Bo Wang[2], Jing Chen[3], Zhiwei Li[1]**
[1]Ocean College, Hebei Agricultural University, Qinhuangdao 066000, Hebei, China
[2]Academic Affairs Office, Hebei University of Environmental Engineering, Qinhuangdao 066102, Hebei, China
[3]Department of Environmental Science, Hebei University of Environmental Engineering, Qinhuangdao 066102, Hebei, China
Corresponding author: Bo Wang (e-mail: linzhenjing207@126.com)

**ABSTRACT** In multi-campus environmental information systems, data silos and privacy risks caused by raw-data sharing hinder collaboration. Similar challenges appear in distributed engineering monitoring, including electromagnetic environmental sensing, where multiple sites must share knowledge without exposing local data or infrastructure details. This paper designs a cross-campus collaborative modeling system using federated learning and differential privacy. The system adopts a hierarchical federated architecture: campuses train models locally and upload only model parameters with calibrated noise to a central server. Dynamic weighting and contribution evaluation mechanisms are integrated to improve generalization under non-IID data distribution while preserving efficient, privacy-protected data collaboration. A hybrid neural network combining one-dimensional convolutional neural networks, gated recurrent units, and an attention mechanism is constructed for environmental time-series characteristics, and gradient-level differential privacy is applied. Experiments using air-quality data from four campuses show that the federated model achieves an 18.1% lower mean absolute error for PM2.5 prediction than independent models. Communication overhead is reduced by 99.1%, and in extreme non-IID scenarios, the error is further reduced by 9.9% compared with standard FedAvg. The results demonstrate improved collaborative modeling for distributed environmental data and provide a transferable architecture for privacy-preserving sensor networks, including electromagnetic monitoring and wireless infrastructure assessment.

**INDEX TERMS** federated learning,privacy protection,non-iid data,distributed sensor networks,electromagnetic monitoring

## I. INTRODUCTION

WITH the advancement of smart campus construction, multi-campus universities have generally deployed distributed environmental information systems, using sensors to continuously collect data such as air quality, temperature, humidity, and noise; simultaneously, the industry is adopting similar sensor networks across multiple factory locations to monitor critical production environments, tracking parameters. However, the physical and administrative isolation between campuses has led to data silos, hindering global environmental analysis and collaborative modeling. This makes it difficult to jointly train high-precision predictive models, thereby limiting cross-campus early warning and governance. While centralizing raw data may seem feasible, it faces challenges such as privacy leakage, data security, and high transmission costs, making it difficult to implement under strict data regulations. Therefore, breaking down data barriers and achieving cross-campus collaborative intelligence while protecting privacy and maintaining local autonomy has become a key issue in multi-campus environmental management.

Traditional data fusion and sharing methods are difficult to effectively solve the data island problem due to privacy risks and high communication costs. In recent years, privacy computing technology, especially federated learning, as a new distributed machine learning paradigm, has provided new ideas for this [1,2]. Its core is to train models locally on each data source and only exchange model parameters rather than original data with the central server, thereby taking into account both privacy protection and collaborative modeling [3-5]. Hu X [6] proposed a federated data sharing solution for the Internet of Vehicles to protect privacy; Li X [7] built a collaborative framework in the vehicle edge network to improve data security; Chen Y [8] combined blockchain with federated learning, and used the transparency and tamper-proof characteristics of blockchain

to enhance privacy protection; Yin L [9] realized privacy-protected multi-party data sharing in the social Internet of Things; Jia B [10] proposed a blockchain-supported federated learning aggregation mechanism, combining differential privacy and homomorphic encryption to enhance security; Lian Z [11] and Zhao Z [12] further proposed an improved federated learning framework to deal with the non-IID data distribution problem. However, the direct application of federated learning to multi-campus environmental information systems still faces specific challenges. Environmental monitoring data exhibits significant temporal correlation and spatial heterogeneity. Due to the varying geographic locations and surrounding environments of various campuses, data distribution often exhibits non-IID characteristics, which severely impacts the convergence and model performance of standard federated learning algorithms. Furthermore, existing research has largely focused on homogeneous data or specific vertical domains. Specific to the heterogeneity of environmental information systems within a multi-campus organization, communication efficiency, and the design of customized federated learning architectures tailored to the specific characteristics of environmental data are still underdeveloped. Therefore, there is an urgent need to design a federated learning system that fully considers the characteristics of environmental data and the needs of multi-campus management to achieve secure and efficient data collaboration.

This paper aims to design and implement a federated learning and data sharing system for multi-campus environmental information systems, providing a foundational, privacy-preserving model transferable to diverse decentralized entities, such as collaborating textile manufacturing plants or regional quality assurance centers. The core work of this paper is to build an end-to-end solution that coordinates the local computing resources of multiple campuses to jointly train a high-quality global environmental prediction model without centralizing any original environmental data. The main contributions of this paper are reflected in three aspects: First, it applies a hierarchical federated learning system architecture suitable for the characteristics of environmental time series data [13,14]. This architecture clarifies the roles, functions and collaboration processes of the central server and the clients on each campus, and provides a basic framework for secure collaborative modeling. Second, based on the basic FedAvg algorithm, an aggregation mechanism based on dynamic contribution evaluation [15] is applied to alleviate model bias caused by uneven data volume and quality across campuses. Differential privacy technology [16,17] is also integrated to inject calibration noise before uploading parameters, thereby defending against privacy inference attacks from a mechanism perspective and achieving dual security. Experiments based on air quality data from four campuses show that the system achieves an approximately 18.1% lower MAE in PM2.5 prediction than the isolated model, with performance approaching that of the centralized model and significantly reduced communication overhead. Furthermore, the prediction error in the non-IID scenario is further reduced by 9.9%, verifying the system's efficiency and security.

The remainder of this paper is organized as follows: Chapter 2 introduces the overall design of the multicampus federated learning system. Chapter 3 focuses on the core technologies and implementation solutions. Chapter 4 presents experiments and results analysis, demonstrating the performance of the proposed method on a real-world dataset. Chapter 5 concludes the research work.

## II. OVERALL ARCHITECTURE DESIGN OF MULTI-CAMPUS FEDERATED LEARNING SYSTEM

### *A. SYSTEM DESIGN GOALS AND CONSTRAINTS*

The primary goal of the system design is to ensure that all raw environmental sensor data is stored on each campus' local server, and model training is performed exclusively locally to ensure data privacy. Furthermore, the interaction between the central server and each campus is strictly limited to deep learning model parameter updates. A differential privacy perturbation mechanism [18,19] is applied to the client side to add precisely controlled noise before parameters are uploaded to further prevent privacy leaks. This improves the learning performance of the global model while effectively protecting privacy. To this end, the second goal of the system design is to use the FedAvg algorithm [20,21] as a collaborative training framework and, combining the characteristics of environmental data with campus differences, design a dynamic contribution evaluation strategy based on campus differences. This strategy dynamically adjusts the aggregation weights of each campus based on its data size and the extent of loss reduction during training. This ensures that campuses with more comprehensive data and more effective training receive a higher proportion of global updates, thereby guiding the model to converge toward better data quality and ultimately achieving performance levels approaching those of centralized training in tasks such as PM2.5 prediction.

Given that federated learning requires multiple rounds of parameter uploads and downloads, communication costs become an important factor affecting overall efficiency. Therefore, the last design goal of the system is to ensure that communication overhead is controllable. On the one hand, after the client executes multiple stochastic gradient descents, only the final updated model differential or complete weight is uploaded, rather than frequently transmitting intermediate gradients, thereby effectively reducing the number of communication rounds. On the other hand, under the premise of ensuring that the accuracy is basically unaffected, the system applies parameter compression technology [22], quantifies the gradient tensor (converts 32-bit floating point numbers to low-precision representation) [23,24] or performs sparse processing (only retains gradients exceeding the set threshold), significantly compressing the amount of data transmitted in a single time. In response to the system heterogeneity caused by differences in hard-

ware performance and network bandwidth between multiple campuses [25], an asynchronous or semi-asynchronous communication mechanism is applied to allow some clients to complete training and participate in aggregation within an acceptable delay range, avoiding the overall training process being hindered by the slow speed of individual nodes. Data heterogeneity refers to the non-IID characteristics that the environmental data of each campus must present. The dynamic contribution evaluation mechanism is one of the means to alleviate this problem. At the same time, the system applies regularization terms during the client-side local training phase to constrain the update direction of the local model so that it does not deviate too much from the global model, thereby preventing "client drift" caused by bias in local data distribution and ensuring the generalization ability of the joint model.

### B. FEDERATED LEARNING SYSTEM ARCHITECTURE

The proposed architecture employs a hierarchical federated learning framework that extends beyond standard FedAvg. While standard FedAvg operates in a single-level star topology, the hierarchical design introduces intermediate coordination mechanisms. Specifically, the hierarchy consists of two levels: (1) the central server level responsible for global model aggregation and coordination, and (2) the campus client level with enhanced local processing capabilities including data preprocessing, local model training with the hybrid architecture, and privacy protection modules. This multi-level structure is particularly advantageous for multicampus environments as it allows for a more efficient and robust division of labor, as shown in Figure 1.

The core idea of the system architecture shown in Figure 1 is to divide model training into two levels: global coordination and local calculation. This mechanism ensures that the original environmental data always remains locally on each campus, and model parameter update information is only transmitted through secure channels.

On the central server side, the scheduling management module uses a node-state-aware adaptive selection algorithm to monitor client computing resources, network latency, and historical participation in real time, dynamically determining the set of nodes participating in each training round. The model distribution module manages global model versions and utilizes differential updates to reduce transmission volume, sending parameter increments only to selected clients. The secure aggregation module performs weighted averaging based on homomorphic encryption, fusing multi-source model parameters in ciphertext to effectively prevent privacy leaks during the aggregation process. The global model repository utilizes a multi-version storage strategy to support rollback operations and performance comparison analysis.

On the client side of each campus, the data preprocessing module performs anomaly detection and correction on environmental sensor data, generating training samples in a unified format through time alignment and sliding window techniques. The local training module initializes the model based on the received global model parameters, performs multiple rounds of optimization using mini-batch gradient descent, and controls the update amplitude through gradient clipping. The privacy protection module injects noise generated by a Gaussian mechanism before uploading parameters, precisely controlling the perturbation intensity based on sensitivity bounds to meet differential privacy requirements.

At the beginning of each training round, the server selects a subset of clients and distributes the latest model. After each node completes local training and privacy processing, it uploads its noisy parameters. After collecting sufficient updates, the server performs secure aggregation and generates a new global model. Through continuous iteration, the system gradually improves global model performance while protecting privacy, enabling efficient collaborative learning of environmental knowledge across campuses.

## III. KEY TECHNOLOGIES AND IMPLEMENTATION

### A. LOCAL MODEL CONSTRUCTION FOR ENVIRONMENTAL DATA

Each campus client uses a hybrid deep learning model combining the Gated Recurrent Unit (GRU) and the one-dimensional convolutional neural network (1D CNN) for local training. This model is specifically designed for the nonlinear temporal characteristics and local patterns of environmental sensor data. It takes as input a preprocessed and standardized feature sequence and uses a sliding window to generate continuous timestep samples.

In this hybrid architecture, 1D CNN uses multi-sized convolutional kernels to extract local short-term features, GRU captures long-term dependencies through a gating mechanism, and the attention mechanism dynamically focuses on key historical moments. These three components work together to address the prediction challenges of short-term correlations, long-term trends, and the suddenness of events, thereby achieving more accurate and robust predictions of PM2.5 concentrations. The front end of the model is a 1D CNN module, which uses a variety of convolutional kernels (lengths 3, 5, and 7) to extract local short-term features along the time dimension [26]. Rectified Linear Unit (ReLU) activation [27,28] and maximum pooling are used to enhance the robustness and expressiveness of the features. The convolutional output is flattened and then input into the GRU layer. The update gate and reset gate mechanism are used to effectively learn long-term dependencies, overcoming the gradient vanishing problem of traditional Recurrent Neural Network (RNN) in long sequence training, and can adaptively memorize important historical information.

An attention mechanism is applied after the GRU layer. Weighted weights are calculated based on the hidden states at different time steps, allowing the model to focus on the most critical historical moments for the current prediction, thereby enhancing its ability to predict sudden environmental changes. The context vector generated by the attention weighting is fed into a fully connected layer and mapped to

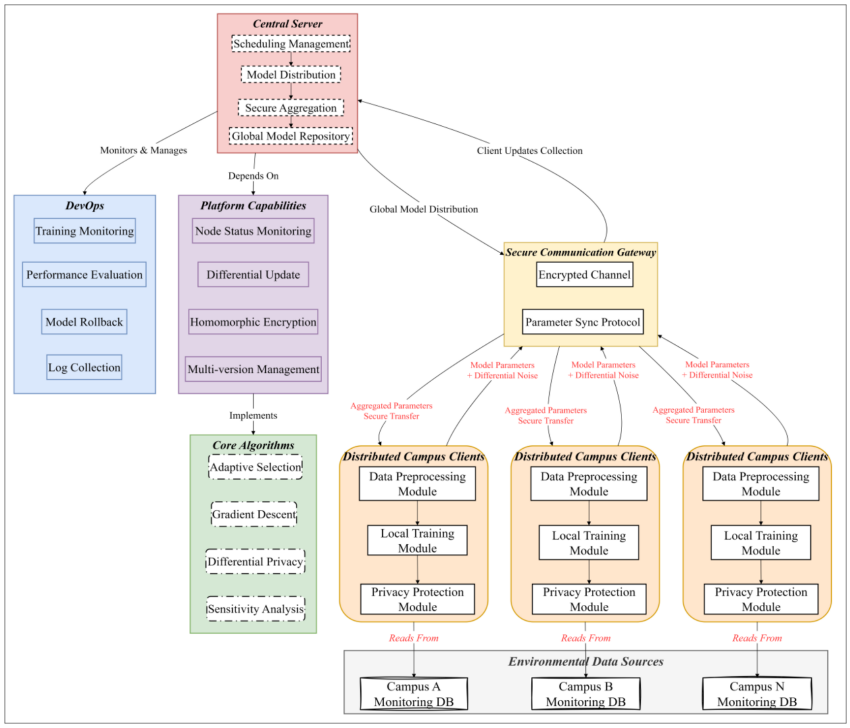


**FIGURE 1.** Federated learning system architecture. Note: DevOps = Development and Operations; DB = database.

the prediction space. The output layer uses a linear activation function to directly regress future PM2.5 concentrations. During training, each client uses the Huber loss function [29,30] for optimization, combining the advantages of mean squared error and mean absolute error (MAE) to improve robustness to outliers. An adaptive moment estimation algorithm is also used to ensure effective convergence of the local model.

### *B. DYNAMIC CONTRIBUTION EVALUATION AGGREGATION MECHANISM FOR NON-IID DATA*

The aggregate weight of each campus client on the global model in each round of federated communication is considered as a dynamic variable. This variable is determined by two key factors: one is the static data volume factor, and the other is the dynamic training loss factor. The data volume factor is used to express that the campus with more data should obtain a greater representation weight in the model [31]. The training loss factor is used to measure the degree of fit and data quality of each campus local model under the current global model state. If a campus obtains a lower loss value after this round of local training, it means that its local data distribution is more consistent with the learning direction of the current global model, and the model parameter update it provides has higher credibility and value, so it should be given a higher weight. On the contrary, if a campus has a higher local loss, it means that its data may have more noise or deviate greatly from the global distribution. At this time, its weight should be appropriately reduced to prevent it from causing negative interference to the global model. Assuming that the local data volume of the k-th campus at the t-th round of communication is $n_k$, and the loss value after local training is $L_k^t$. First, the data volume is normalized to obtain the data volume factor:

$$\alpha_k = \frac{n_k}{\sum_{i=1}^{N} n_i} \tag{1}$$

where $N$ is the total number of campuses participating in the training. Secondly, the loss factor is calculated based on the loss value set of all campuses participating in the training in this round. To convert the loss value into a positive contribution index, a normalized exponential function is used for calculation. First, the inverse of the loss value of each campus is calculated, $1/L_k^t$, and then the softmax function [32] is applied for normalization to obtain the loss factor:

$$\beta_k^t = \frac{\exp(1/L_k^t)}{\sum_{i=1}^{N} \exp(1/L_i^t)} \tag{2}$$

This process results in campuses with lower losses having larger $\beta_k^t$ values, indicating a higher contribution. Finally, the aggregate weight $\omega_k^t$ of the k-th campus in round t is calculated by taking the weighted geometric mean of the data volume factor $\alpha_k$ and the loss factor $\beta_k^t$, i.e.:

$$\omega_k^t = \left(\alpha_k^{\lambda} \cdot (\beta_k^t)^{1-\lambda}\right)^{1/2} \tag{3}$$

$\lambda$ is a hyperparameter between 0 and 1 that balances the relative importance of data volume and training loss in weight allocation. When $\lambda = 1$, the mechanism degenerates into the standard FedAvg algorithm [33,34]. When $\lambda = 0$,

weight allocation is completely dependent on training loss. By adjusting $\lambda$, it can be flexibly adapted to different application scenarios. After obtaining the dynamic weights of each campus, they are normalized to ensure $\sum_{k=1}^{N} \omega_k^t = 1$. Subsequently, the central server performs weighted aggregation and updates the global model parameters:

$$\omega_{t+1} = \sum_{k=1}^{N} \omega_k^t \cdot \omega_{t+1}^k \tag{4}$$

where $\omega_{t+1}^k$ is the local model parameter uploaded by the k-th campus. The workflow of this dynamic contribution evaluation aggregation mechanism is shown in Figure 2.

Figure 2 illustrates the application of the dynamic loss factor in the system. By applying this factor, the system can adaptively identify high-quality model updates and give them greater weight, while weakening the impact of low-quality or abnormal updates.

### C. MODEL UPDATES PROTECTION ENHANCED BY DIFFERENTIAL PRIVACY

Differential privacy technology is applied before uploading local model parameters to add precisely calibrated random noise to the model updates. This protects the privacy of individual data records while retaining the overall statistical properties of the gradient as much as possible to ensure effective aggregation and convergence of the global model. Using a differential privacy scheme based on the Gaussian mechanism [35], after each round of local training of federated learning is completed, each campus client needs to preprocess the model gradients it has calculated. The first step is gradient clipping, which limits the upper limit of each sample's contribution to the overall gradient, thereby controlling the sensitivity of the gradient. For a campus with a local dataset size of $n_k$, its local model generates a set of gradients $g$. This paper performs L2 norm clipping: for each sample calculated gradient $g_i$, check its norm $\|g_i\|_2$. If the norm exceeds the preset clipping threshold $C$, the gradient is scaled to:

$$g_i = g_i \cdot C \|g_i\|_2 \tag{5}$$

This ensures that the clipped gradient satisfies:

$$\|g_i\|_2 \leq C \tag{6}$$

After gradient clipping, the next step is to add Gaussian noise to the aggregated batch gradients. The client calculates the average gradient of all samples in this round:

$$\bar{g} = \frac{1}{n_k} \sum_{i=1}^{n_k} g_i \tag{7}$$

Subsequently, noise is generated based on the preset privacy parameters. The protection strength of differential privacy $\epsilon$ is defined by $\delta$ the tolerance parameter and $\delta$ is typically set to a very small value, representing the maximum probability of privacy failure. The gradient sensitivity is determined $\Delta f$ by $C$ and $n_k$. For the average gradient, its sensitivity is $\Delta f = C/n_k$. Therefore, the required noise standard deviation $\sigma$ is related to $\Delta f$, $\epsilon$, and $\delta$, and is calculated as:

$$\sigma = \frac{\Delta f \cdot \sqrt{2 \ln(1.25/\delta)}}{\epsilon} \tag{8}$$

The client samples a noise vector $N(0, \sigma^2 I)$ from a Gaussian distribution with mean 0 and standard deviation $\sigma$ and adds it to the average gradient to obtain the protected gradient

$$\tilde{g} = \bar{g} + N(0, \sigma^2 I) \tag{9}$$

Finally, the client uploads the noisy model updates $\tilde{g}$ to the central server instead of the original model parameters or gradients.

## IV. EXPERIMENTAL VERIFICATION AND ANALYSIS

### A. EXPERIMENTAL SETUP AND DATASET

An air quality monitoring dataset for four campuses (Campus A, B, C, and D) is constructed based on data collected by a network of environmental sensors deployed on each campus. The data is collected from January 1, 2022, to December 31, 2022, with an hourly sampling frequency. During data preprocessing, linear interpolation is used to address missing values due to temporary sensor failures, and all feature data is Zscore normalized to accelerate model training convergence. Detailed statistical characteristics of the dataset are shown in Table 1.

**TABLE 1.** Statistical characteristics of the multi-campus air quality monitoring dataset

| Campus | Total data entries | PM2.5 concentration ($\mu$g/m3) Mean | SD | Mean temperature ($^{\circ}$C) | Mean humidity (%) |
|---|---|---|---|---|---|
| A | 8215 | 38.7 | 22.4 | 18.5 | 65.2 |
| B | 7542 | 45.2 | 28.1 | 17.8 | 68.9 |
| C | 6830 | 32.5 | 18.9 | 19.2 | 62.1 |
| D | 5988 | 41.6 | 25.3 | 16.9 | 71.5 |

Note: SD = standard deviation; the total data entries column is reported in number of records.

Table 1 shows significant differences in data volume among Campuses A, B, C, and D. Campus A has the largest data set (8,215 records), while Campus D has the smallest (5,988 records). This reflects real-world variations in monitoring infrastructure and operational time across campuses. The distribution of PM2.5 concentrations, the target variable, also exhibits significant heterogeneity in mean and SD values across campuses. Campus B has the highest mean concentration (45.2 $\mu$g/m$^3$) and the greatest volatility (SD = 28.1 $\mu$g/m$^3$), while Campus C has the best and most stable environmental quality (mean 32.5 $\mu$g/m$^3$, SD = 18.9 $\mu$g/m$^3$).

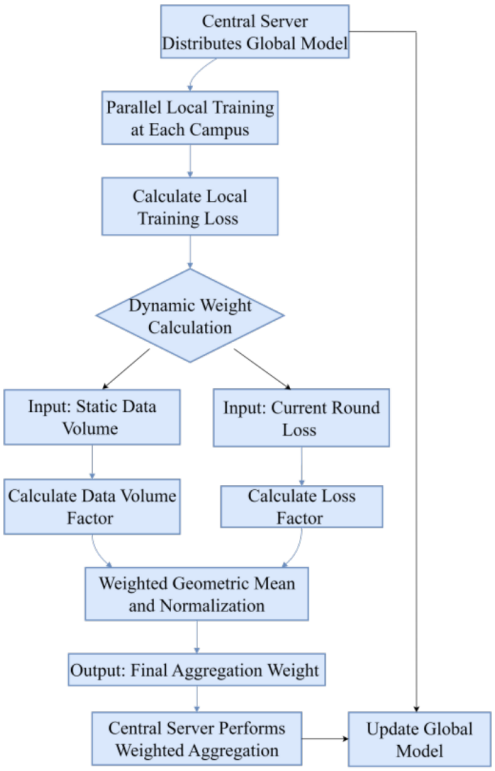


**FIGURE 2.** Federated aggregation and dynamic weight calculation

The experiment sets up three baseline models for performance comparison. The first is the independent training model M1 for each campus, that is, each campus only uses its own local data for training and prediction, and does not participate in any form of collaboration. This baseline is used to evaluate the lower limit of model performance under data silos. The second is the traditional centralized training model M2, which assumes that the raw data of all campuses can be unconditionally aggregated into a central data center for unified training. This baseline represents the theoretical upper limit of performance. The third is the standard FedAvg (M4), which uses a uniformly weighted traditional federated averaging algorithm as the benchmark federated learning method. The federated learning system M3 used in this article will be compared with these baselines on the same test set. All comparison models are based on the same Long Short-Term Memory (LSTM) network structure and are trained under exactly the same hyperparameter settings to ensure fairness in the comparison.

### *B. COMPARATIVE ANALYSIS OF MODEL PREDICTION PERFORMANCE*

The PM2.5 concentration prediction performance of M1, M2, and M3 is compared on a unified test set. MAE and root mean square error (RMSE), widely used in regression tasks, are used as core metrics. Lower values indicate higher prediction accuracy. The quantitative results are shown in Figure 3.

Figure 3 shows the prediction errors of the independently trained models for each campus are significantly higher than those of the other models. Specifically, as shown in Figures 3(a) and 3(b), the average MAE and RMSE of the four independent models for the four campuses are 9.85 $\mu$g/m$^3$ and 13.72 $\mu$g/m$^3$, respectively. In contrast, the federated learning system, through secure parameter aggregation, effectively leverages cross-campus data patterns, reducing the average MAE to 8.07 $\mu$g/m$^3$ (a decrease of approximately 18.1% relative to M1) and the average RMSE to 11.24 $\mu$g/m$^3$. Furthermore, the performance of the federated learning system is very close to the ideal baseline for centralized training (with an average MAE of 7.63 $\mu$g/m$^3$ and an average RMSE of 10.61 $\mu$g/m3), with the difference being within an acceptable range. This demonstrates that federated learning achieves data privacy protection with little to no loss of accuracy, confirming its effectiveness in breaking down data silos.

The comparison between the predicted values of each model and the true values within a continuous 72-hour period of the test set is shown in Figure 4.

Figure 4 shows a comparison of the PM2.5 concentration prediction performance of M1 and M3. Figure 4(a) shows the 72-hour monitoring period on the horizontal axis and the PM2.5 concentration on the vertical axis. The solid black line, representing the actual monitored value, reaches its peak concentration at approximately 30 hours. The M1 prediction trajectory exhibits significant overall deviation, with significant lag and underestimation of the peak value. The M2 prediction almost completely overlaps with the ac-

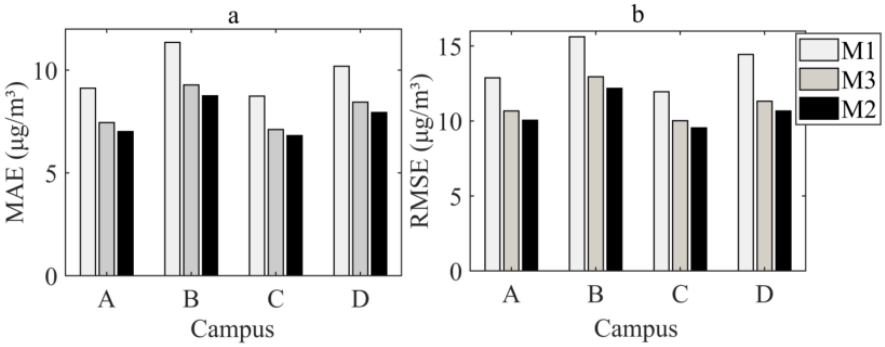


**FIGURE 3.** Comparison of MAE and RMSE of different models on the test set. Figure 3(a). MAE comparison; Figure 3(b). RMSE comparison

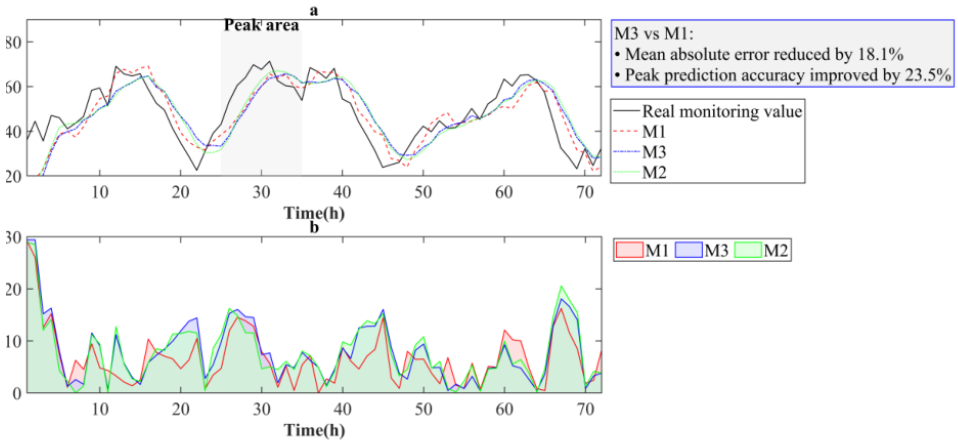


**FIGURE 4.** Comparative analysis of PM2.5 concentration prediction performance of multiple models. Figure 4 (a). Comparative analysis of PM2.5 concentration prediction performance of multi-campus environmental information systems; Figure 4 (b). Time series analysis of prediction errors of each model

tual value. The M3 prediction trajectory closely matches the actual value, accurately capturing the phase and amplitude of concentration changes.

Figure 4(b) shows the absolute error of each model over time. The independently trained model exhibits high and fluctuating errors, with large peak errors. The federated learning system maintains a low and stable error, with performance approaching that of the centralized training model. This demonstrates that, while ensuring data privacy, the federated learning system can achieve prediction accuracy and robustness comparable to centralized training, demonstrating application reliability.

Finally, a comprehensive comparison was made between M4 and M1~M3, highlighting the overall performance of M3. The results are shown in the Table 2 below.

**TABLE 2.** Comprehensive performance comparison of all methods

| Method | MAE ($\mu$g/m3) | RMSE ($\mu$g/m3) | Performance gap to centralized |
|---|---|---|---|
| M1 | 9.85 | 13.72 | 29.1% |
| M4 | 8.52 | 11.89 | 11.7% |
| M3 | 8.07 | 11.24 | 5.8% |
| M2 | 7.63 | 10.61 | 0% |

As shown in Table 2, M3 outperforms the standard FedAvg baseline model (M4, 8.52 $\mu$g/m$^3$). Although the centralized model (M2, 7.63 $\mu$g/m$^3$) remains the theoretically optimal solution, our proposed method narrows the performance gap between independent and centralized training, representing a significant i mprovement o ver standard federated learning methods.

### *C. ROBUSTNESS ANALYSIS UNDER HETEROGENEOUS DATA DISTRIBUTION*

Based on the quartiles of PM2.5 concentrations in the original dataset, the data is divided into four concentration ranges (0-25%, 25-50%, 50-75%, and 75-100%). Each campus is forced to contain data from only one specific

concentration range, resulting in four clients with distinct data distributions. Under this extreme non-IID setting, the performance of the standard FedAvg algorithm is compared with the dynamic contribution evaluation aggregation mechanism used in this paper. The training process lasted 80 communication rounds, and the MAE value of the global model on a unified test set covering all concentration ranges is recorded after each round. The performance comparison results are shown in Figure 5.

Figure 5 treats all client model updates equally, failing to fully consider the impact of data distribution deviations across campuses on the global model. This results in significant oscillations during convergence and ultimately stabilizes at a high MAE (9.35 $\mu g/m^3$). In contrast, the dynamic contribution evaluation mechanism adjusts aggregation weights by evaluating the training loss of each campus in real time. This adaptively reduces the influence of campuses with anomalous data distributions while simultaneously increasing the contribution of campuses with better data quality. This results in a smoother convergence process and reduces the final MAE to 8.42 $\mu g/m^3$, a performance improvement of approximately 9.9%.

The weight distribution of each campus under the dynamic contribution evaluation mechanism during the 40th round of training is shown in Table 3.

**TABLE 3.** Weight distribution of different campuses under the dynamic contribution evaluation mechanism

| Campus | Data volume (items) | Local training loss | Standard FedAvg weight | Dynamic contribution evaluation weight |
|---|---|---|---|---|
| A | 2150 | 0.134 | 0.28 | 0.26 |
| B | 2890 | 0.156 | 0.38 | 0.24 |
| C | 1240 | 0.089 | 0.16 | 0.31 |
| D | 1980 | 0.121 | 0.26 | 0.29 |

Table 3 shows that in the standard FedAvg model, where weights are determined solely by data volume, Campus B receives the highest weight of 0.38 due to its large data volume, even though its data only represents high-PM2.5 concentration ranges. However, in the dynamic contribution evaluation mechanism, Campus B's weight is dynamically adjusted to 0.24 due to its highest local training loss (0.156). Conversely, Campus C, despite having the smallest data volume, receives a relatively high weight of 0.31 due to its lowest training loss (0.089). Overall, the data demonstrate that the dynamic weight adjustment mechanism based on data quality effectively mitigates the negative impact of non-IID data on the global model, demonstrating its robustness against data heterogeneity in real-world multi-campus environmental monitoring scenarios.

### D. ABLATION STUDY ON DYNAMIC WEIGHTING MECHANISM

To quantitatively evaluate the contribution of the proposed dynamic weighting mechanism, an ablation experiment was conducted, comparing three configurations: (1) the standard FedAvg algorithm with uniform weighting; (2) weighting based solely on data volume; and (3) the proposed dynamic weighting mechanism combining data volume and training loss factor. The results are summarized in Table 4.

**TABLE 4.** Ablation study results comparing different weighting strategies

| Weighting strategy | MAE ($\mu g/m3$) | RMSE ($\mu g/m3$) | Convergence rounds |
|---|---|---|---|
| Uniform (FedAvg) | 9.35 | 13.12 | 78 |
| Data volume only | 8.87 | 12.45 | 72 |
| Dynamic weighting | 8.42 | 11.89 | 65 |

The results of Table 4 showing that the complete dynamic weighting mechanism achieves the best performance in the non-independent identically distributed (non-IID) scenario.

### E. ANALYSIS OF PRIVACY PROTECTION EFFECT

With $\delta$ fixed at $10^{-15}$, $\epsilon$ is set to 0.1, 0.5, 1.0, 2.0, 5.0, and infinity (corresponding to the baseline federated learning scenario with no noise addition) to explore the quantitative tradeoff between privacy protection strength and model utility. For each ($\epsilon$, $\delta$) pair, the full federated learning training process is re-executed according to the algorithm, and the MAE of the final model on the unified test set is recorded. The tradeoff between differential privacy protection strength and model prediction accuracy is shown in Figure 6.

Figure 6 shows the MAE for PM2.5 prediction. As $\epsilon$ the MAE increases from 0.1 to 5.0, the model MAE decreases continuously from 9.52 to 8.15, indicating that weakening the privacy protection strength leads to a steady improvement in model prediction accuracy. When privacy protection is completely eliminated ($\epsilon = \infty$), the MAE reaches its lowest value of 8.07. This change clearly demonstrates that stronger privacy protection (low values) in the differential privacy mechanism results in a certain loss in model performance, but appropriate adjustments $\epsilon$ can achieve a reasonable balance between privacy protection and model accuracy.

### F. ANALYSIS OF SYSTEM CONVERGENCE AND COMMUNICATION EFFICIENCY

The analysis is conducted from two dimensions: system convergence and communication efficiency. In the convergence analysis, the loss changes of M3 and M2 are recorded during the training process under the same test set loss. The training process lasted for 100 communication rounds. After each round, the MAE of the global model on the validation set is calculated on the central server. The convergence analysis results are shown in Figure 7.

Figure 7 shows that the federated learning system exhibits rapid loss reduction in the initial phase (first 20 epochs), with the slope of its convergence curve roughly comparable to that of the centralized approach. As training progresses, the federated learning system's convergence rate gradually

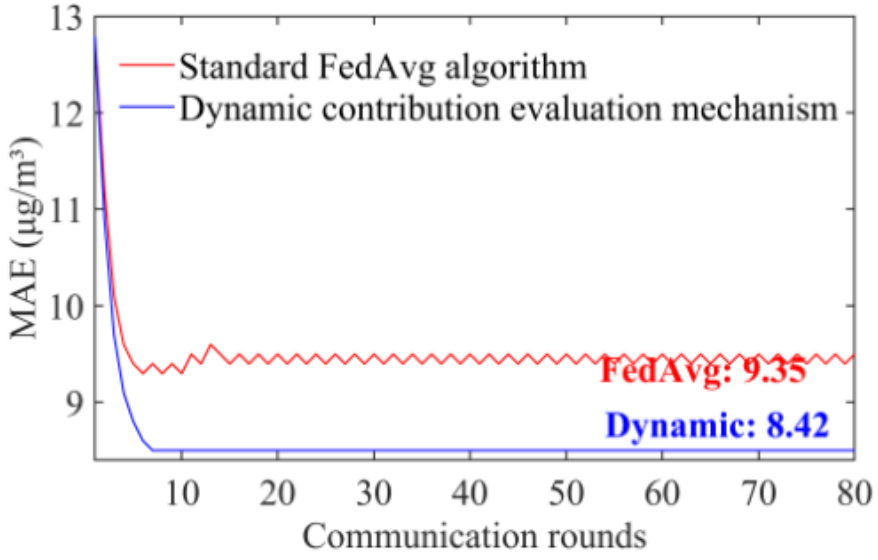


**FIGURE 5.** Performance comparison of different aggregation algorithms for non-IID data

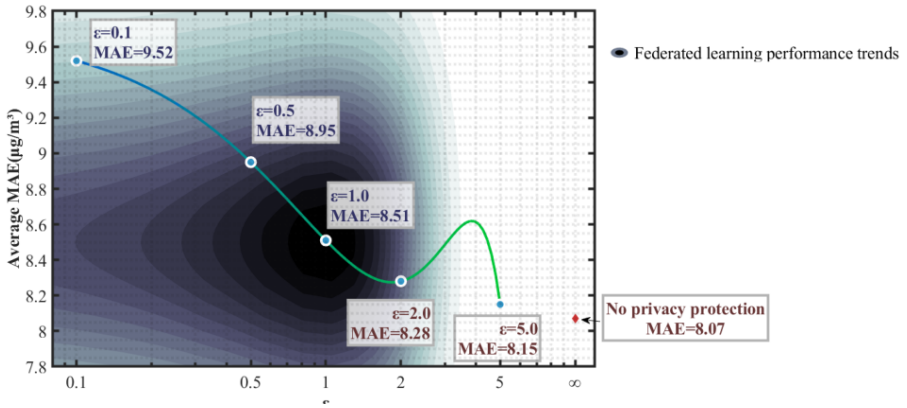


**FIGURE 6.** Trade-off between differential privacy protection strength and model prediction accuracy

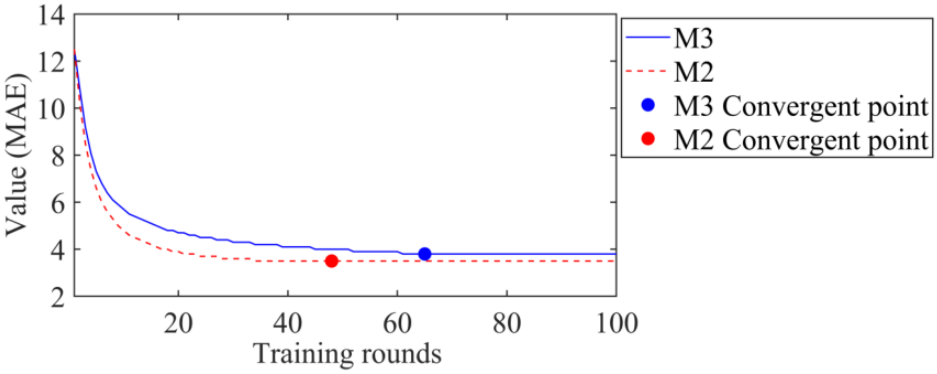


**FIGURE 7.** Convergence curve comparison between the federated learning system and the centralized training model (unit: $\mu$g/m3)

slows, but it reaches the target loss value by the 65th communication epoch, while the centralized training model achieves the same performance level by the 48th training epoch. Although the federated learning system requires approximately 35% more iterations to converge than the centralized training model, its final stabilized test loss is similar to the centralized training model baseline, highlighting its excellent convergence stability.

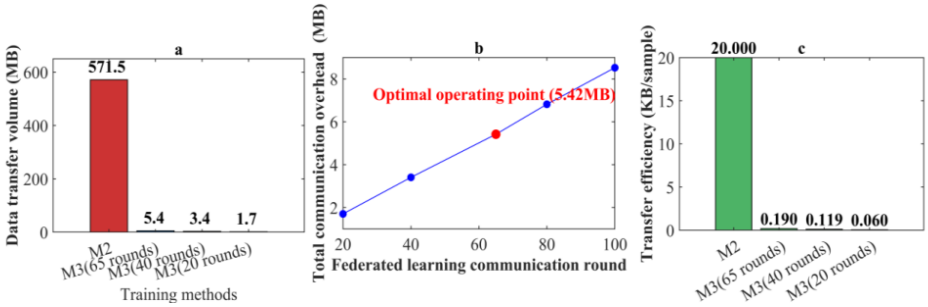


**FIGURE 8.** Comprehensive evaluation of communication efficiency. Figure 8 (a). Comparative analysis of communication overhead; Figure 8 (b). Relationship between communication rounds and overhead; Figure 8 (c). Comparison of data transmission efficiency per unit sample

In the communication efficiency analysis, the total amount of data transmission required to achieve the target test loss for both training methods (federated learning and centralized training) is precisely calculated. The federated learning training method accumulated the model parameter sizes uploaded by all participating campuses in each round, while the centralized training method calculated the total amount of feature data for all samples in the original environmental monitoring dataset. The communication efficiency comparison results are shown in Figure 8.

Figure 8 comprehensively compares the communication efficiency of federated learning and centralized training through three sub-figures. Figure 8 (a) shows that centralized training requires the transmission of 571.5 MB of raw data, while the amount of data transmitted by federated learning decreases linearly with the number of rounds, highlighting that federated learning can save 99.1% to 99.7% of communication overhead. Figure 8 (b) shows the proportional relationship between rounds and overhead (1.7 MB for 20 rounds to 8.5 MB for 100 rounds), and marks the optimal operating point at 65 rounds (5.42 MB), demonstrating that this system achieves maximum communication efficiency while maintaining model performance. Figure 8 (c) shows that the centralized training method has the lowest efficiency (20.0 KB/sample), while the efficiency of federated learning increases significantly at different rounds (0.190 KB/sample for 65 rounds and 0.060 KB/sample for 20 rounds). These changes demonstrate that federated learning, by replacing raw data sharing with parameter transmission, achieves orders of magnitude reduction in communication overhead in multicampus environments, providing an efficient collaborative solution for distributed environmental monitoring systems.

## V. CONCLUSION

This paper successfully designs and validates a federated learning and data sharing system for a multi-campus environmental information system, effectively resolving the core conflict between data silos and privacy security, a solution which establishes a clear methodology for manufacturing groups to enable collaborative, cross-factory data intelligence without exposing proprietary operational details. Experimental results demonstrate that, without centralizing any raw data, the PM2.5 prediction model implemented through federated learning significantly outperforms isolated models on each campus (MAE reduction of approximately 18.1%) and approaches the theoretical upper limit of centralized training. By applying differential privacy, the system achieves a good balance between privacy protection and model utility (MAE = 8.51 $\mu g/m^3$) with a privacy budget of 1. Furthermore, the system only requires transmitting model parameters, reducing communication overhead by at least 99.1% compared to centralized transmission of raw data. Combined with a dynamic contribution evaluation mechanism, the system demonstrates enhanced robustness in heterogeneous data environments. This study demonstrates that federated learning is an effective approach to breaking down data barriers and achieving secure collaborative intelligence, providing a viable solution for multi-campus environmental management and offering a secure paradigm for industry conglomerates seeking to unify quality control or resource optimization across geographically disparate manufacturing facilities.

## AUTHOR CONTRIBUTIONS

Conceptualization – Zhenjing Lin, Bo Wang, Jing Chen and Zhiwei Li; methodology – Zhenjing Lin, Bo Wang, Jing Chen and Zhiwei Li; investigation – Zhenjing Lin, Bo Wang and Zhiwei Li; writing-original draft preparation – Zhenjing Lin, Bo Wang, Jing Chen and Zhiwei Li. All authors have read and agreed to the published version of the manuscript.

## CONFLICTS OF INTEREST

The authors declare no conflict of interest.

## FUNDING

This research was supported by,
1. Hebei Province higher education teaching reform research and practice project(2022GJJG381)

## ACKNOWLEDGEMENTS

Not applicable.

## REFERENCES

[1] J. Liu, J. Huang, Y. Zhou, X. Li, S. Ji, H. Xiong, et al., "From distributed machine learning to federated learning: A survey," Knowledge and information systems, vol. 64, no. 4, pp. 885-917, 2022, doi: 10.1007/s10115-022-01664-x.

[2] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. V. Poor, "Federated learning for internet of things: A comprehensive survey," IEEE communications & tutorials, vol. 23, no. 3, pp. 1622-1658, 2021, doi: 10.1109/COMST.2021.3075439.

[3] D. C. Nguyen, M. Ding, Q. V. Pham, P. N. Pathirana, L. B. Le, A. Seneviratne, et al., "Federated learning meets blockchain in edge computing: Opportunities and challenges," IEEE Internet of Things Journal, vol. 8, no. 16, pp. 12806-12825, 2021, doi: 10.1109/JIOT.2021.3072611.

[4] J. Wen, Z. Zhang, Y. Lan, Z. Cui, J. Cai, and W. Zhang, "A survey on federated learning: challenges and applications," International journal of machine learning and cybernetics, vol. 14, no. 2, pp. 513-535, 2023, doi: 10.1007/s13042-022-01647-y.

[5] Q. Li, Z. Wen, Z. Wu, S. Hu, N. Wang, Y. Li, et al., "A survey on federated learning systems: Vision, hype and reality for data privacy and protection," IEEE Transactions on Knowledge and Data Engineering, vol. 35, no. 4, pp. 3347-3366, 2021, doi: 10.1109/TKDE.2021.3124599.

[6] X. Hu, R. Li, L. Wang, Y. Ning, and K. Ota, "A data sharing scheme based on federated learning in IoV," IEEE Transactions on Vehicular Technology, vol. 72, no. 9, pp. 11644-11656, 2023, doi: 10.1109/TVT.2023.3266100.

[7] X. Li, L. Cheng, C. Sun, K. Y. Lam, X. Wang, and F. Li, "Federated-learning-empowered collaborative data sharing for vehicular edge networks," IEEE network, vol. 35, no. 3, pp. 116-124, 2021, doi: 10.1109/MNET.011.2000558.

[8] Y. Chen, J. Li, F. Wang, K. Yue, Y. Li, B. Xing, et al., "DS2PM: A data-sharing privacy protection model based on blockchain and federated learning," IEEE Internet of Things Journal, vol. 10, no. 14, pp. 12112-12125, 2021, doi: 10.1109/JIOT.2021.3134755.

[9] L. Yin, J. Feng, H. Xun, Z. Sun, and X. Cheng, "A privacy-preserving federated learning for multiparty data sharing in social IoTs," IEEE Transactions on Network Science and Engineering, vol. 8, no. 3, pp. 2706-2718, 2021, doi: 10.1109/TNSE.2021.3074185.

[10] B. Jia, X. Zhang, J. Liu, Y. Zhang, K. Huang, and Y. Liang, "Blockchain-enabled federated learning data protection aggregation scheme with differential privacy and homomorphic encryption in IIoT," IEEE Transactions on Industrial Informatics, vol. 18, no. 6, pp. 4049-4058, 2021, doi: 10.1109/TII.2021.3085960.

[11] Z. Lian, Q. Zeng, W. Wang, T. R. Gadekallu, and C. Su, "Blockchain-based two-stage federated learning with non-IID data in IoMT system," IEEE Transactions on Computational Social Systems, vol. 10, no. 4, pp. 1701-1710, 2022, doi: 10.1109/TCSS.2022.3216802.

[12] Z. Zhao, C. Feng, W. Hong, J. Jiang, C. Jia, T. Q. Quek, et al., "Federated learning with non-IID data in wireless networks," IEEE Transactions on Wireless communications, vol. 21, no. 3, pp. 1927-1942, 2021, doi: 10.1109/TWC.2021.3108197.

[13] H. Zheng, M. Gao, Z. Chen, and X. Feng, "A distributed hierarchical deep computation model for federated learning in edge computing," IEEE Transactions on Industrial Informatics, vol. 17, no. 12, pp. 7946-7956, 2021, doi: 10.1109/TII.2021.3065719.

[14] Z. Wang, H. Xu, J. Liu, Y. Xu, H. Huang, and Y. Zhao, "Accelerating federated learning with cluster construction and hierarchical aggregation," IEEE Transactions on Mobile Computing, vol. 22, no. 7, pp. 3805-3822, 2022, doi: 10.1109/TMC.2022.3147792.

[15] X. Tu, K. Zhu, N. C. Luong, D. Niyato, Y. Zhang, and J. Li, "Incentive mechanisms for federated learning: From economic and game theoretic perspective," IEEE transactions on cognitive communications and networking, vol. 8, no. 3, pp. 1566-1593, 2022, doi: 10.1109/TCCN.2022.3177522.

[16] Y. Zhao and J. Chen, "A survey on differential privacy for unstructured data content," ACM Computing Surveys (CSUR), vol. 54, no. 10s, pp. 1-28, 2022, doi: 10.1145/3490237.

[17] B. Jiang, J. Li, G. Yue, and H. Song, "Differential privacy for industrial internet of things: Opportunities, applications, and challenges," IEEE Internet of Things Journal, vol. 8, no. 13, pp. 10430-10451, 2021, doi: 10.1109/JIOT.2021.3057419.

[18] B. Wang, Y. Chen, H. Jiang, and Z. Zhao, "Ppefl: Privacy-preserving edge federated learning with local differential privacy," IEEE Internet of Things Journal, vol. 10, no. 17, pp. 15488-15500, 2023, doi: 10.1109/JIOT.2023.3264259.

[19] R. Hu, Y. Guo, and Y. Gong, "Federated learning with sparsified model perturbation: Improving accuracy under client-level differential privacy," IEEE Transactions on Mobile Computing, vol. 23, no. 8, pp. 8242-8255, 2023, doi: 10.1109/TMC.2023.3343288.

[20] T. Sun, D. Li, and B. Wang, "Decentralized federated averaging," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 45, no. 4, pp. 4289-4301, 2022, doi: 10.1109/TPAMI.2022.3196503.

[21] J. Liu, J. Lou, L. Xiong, J. Liu, and X. Meng, "Projected federated averaging with heterogeneous differential privacy," Proceedings of the VLDB Endowment; 16–20 August 2021; Copenhagen, Denmark 2021; 15(4):828-840, doi: 10.14778/3503585.3503592.

[22] P. V. Dantas, W. Sabino da Silva Jr, L. C. Cordeiro, L. C. Cordeiro, and C. B. Carvalho, "A comprehensive review of model compression techniques in machine learning," Applied Intelligence, vol. 54, no. 22, pp. 11804-11844, 2024, doi: 10.1007/s10489-024-05747-w.

[23] C. Y. Lin, V. Kostina, and B. Hassibi, "Differentially quantized gradient methods," IEEE Transactions on Information Theory, vol. 68, no. 9, pp. 6078-6097, 2022, doi: 10.1109/TIT.2022.3171173.

[24] S. Y. Chang and H. C. Wu, "Tensor quantization: High-dimensional data compression," IEEE Transactions on Circuits and Systems for Video Technology, vol. 32, no. 8, pp. 5566-5580, 2022, doi: 10.1109/TCSVT.2022.3145341.

[25] M. Ye, X. Fang, B. Du, P. C. Yuen, and D. Tao, "Heterogeneous federated learning: State-of-the-art and research challenges," ACM Computing Surveys, vol. 56, no. 3, pp. 1-44, 2023, doi: 10.1145/3625558.

[26] Y. Liao, Y. Wang, X. Zeng, M. Wu, and X. Qing, "Multiscale 1-DCNN for damage localization and quantification using guided waves with novel data fusion technique and new self-attention module," IEEE Transactions on Industrial Informatics, vol. 20, no. 1, pp. 492-502, 2023, doi: 10.1109/TII.2023.3268442.

[27] S. Zhang, J. Lu, and H. Zhao, "Deep network approximation: Beyond relu to diverse activation functions," Journal of Machine Learning Research, vol. 25, no. 35, pp. 1-39, 2024, doi: 10.48550/arXiv.2307.06555.

[28] S. Kılıçarslan, K. Adem, and M. Çelik, "An overview of the activation functions used in deep learning algorithms," Journal of New Results in Science, vol. 10, no. 3, pp. 75-88, 2021, doi: 10.54187/jnrs.1011739.

[29] J. Xie, S. Liu, J. Chen, and J. Jia, "Huber loss based distributed robust learning algorithm for random vector functional-link network," Artificial Intelligence Review, vol. 56, no. 8, pp. 8197-8218, 2023, doi: 10.1007/s10462-022-10362-7.

[30] W. Sun, K. Wang, M. Wang, and Y. Song, "L21 Norm Regularization ELM with p-Huber Loss Function for Multitarget Regression," IAENG International Journal of Applied Mathematics, vol. 55, no. 8, pp. 2570-2580, 2025, doi: 10.1109/ACCESS.2018.2887260.

[31] J. Pei, W. Liu, J. Li, L. Wang, and C. Liu, "A review of federated learning methods in heterogeneous scenarios," IEEE Transactions on Consumer Electronics, vol. 70, no. 3, pp. 5983-5999, 2024, doi: 10.1109/TCE.2024.3385440.

[32] K. Chen, Y. Gao, H. Waris, W. Liu, and F. Lombardi, "Approximate softmax functions for energy-efficient deep neural networks," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 31, no. 1, pp. 4-16, 2022, doi: 10.1109/TVLSI.2022.3224011.

[33] Y. Zhou, Q. Ye, and J. Lv, "Communication-efficient federated learning with compensated overlap-fedavg," IEEE Transactions on Parallel and Distributed Systems, vol. 33, no. 1, pp. 192-205, 2021, doi: 10.1109/TPDS.2021.3090331.

[34] L. Su, J. Xu, and P. Yang, "A non-parametric view of fedavg and fedprox: Beyond stationary points," Journal of Machine Learning Research, vol. 24, no. 203, pp. 1-48, 2023, doi: 10.48550/arXiv.2106.15216.

[35] J. Dong, A. Roth, and W. J. Su, "Gaussian differential privacy," Journal of the Royal Statistical Society Series B: Statistical Methodology, vol. 84, no. 1, pp. 3-37, 2022, doi: 10.1111/rssb.12454.