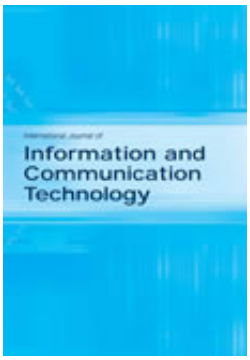
INDERSCIENCE PUBLISHERS
Linking academia, business and industry through research



International Journal of Information and Communication Technology

ISSN online: 1741-8070 - ISSN print: 1466-6642
https://www.inderscience.com/ijict

Research on the optimisation of communication efficiency based on adaptive improved federated learning

Xuefei Zhang, Yanli Zhao

DOI: 10.1504/IJICT.2025.10071524

Article History:

Received:	10 December 2024
Last revised:	26 March 2025
Accepted:	26 March 2025
Published online:	18 June 2025

Research on the optimisation of communication efficiency based on adaptive improved federated learning

Xuefei Zhang*

Department of Mechanical Engineering,
Shanxi Engineering Vocational College,
Taiyuan, 030000, China
Email: 15235398536@163.com
*Corresponding author

Yanli Zhao

China Mobile Communication Group Design Institute Co., Ltd.,
Shanxi Branch,
Taiyuan, 030000, China
Email: zhaoyanli@cmdi.chinamobile.com

Abstract: Aiming at the communication efficiency bottleneck in the internet of things and edge computing scenarios, this paper proposes a communication efficiency improvement scheme based on adaptive improved federated learning. By constructing an ARMA bandwidth prediction model enhanced by wavelet transform, the client network environment is predicted, and the improved Sketch compression algorithm is adopted to dynamically adapt to the real-time bandwidth conditions, thus the communication efficiency optimisation in the internet of things and edge computing scenarios is achieved. Experiments show that the accuracy of the proposed method researches 95%, the average uplink communication time is 0.5 seconds, and the communication efficiency exceeds 1.7. It provides key technical support for real-time federated learning deployment in 5G edge computing environment.

Keywords: federated learning; wavelet transform; ARMA; Sketch; communication efficiency.

Reference to this paper should be made as follows: Zhang, X. and Zhao, Y. (2025) 'Research on the optimisation of communication efficiency based on adaptive improved federated learning', *Int. J. Information and Communication Technology*, Vol. 26, No. 20, pp.19–40.

Biographical notes: Xuefei Zhang studied at the Inner Mongolia Normal University from 2012 to 2016, and obtained a Bachelor's degree in 2016. She studied at Chang'an University from 2016 to 2019 and obtained a Master's degree in 2019. Currently, she works in Shanxi Engineering Vocational College and also published five papers. Her research interests include distributed storage, network coding and local repair code.

Yanli Zhao studied in North University of China from 2001 to 2005, and received his Bachelor's degree in 2005. She studied in North University of China from 2006 to 2009, and obtained a Master's degree in 2009. Currently,

she works in Shanxi Branch of China Mobile Communications Group Design Institute Co., Ltd. and also published four papers. His research interests include 5G core network networking technology and mobile cloud technology.

1 Introduction

Federated learning, as an important paradigm for distributed machine learning, shows significant potential in edge computing and IoT, but its communication efficiency bottleneck needs to be broken. Existing research revolves around federated learning framework optimisation, model compression techniques, and dynamic network adaptation, but there are still significant limitations in each direction. This article reveals research gaps through a systematic literature review and proposes innovative solutions.

Regarding the federated learning optimisation of the IIoT scenario, Li et al. proposed the FSLEdge framework to reduce the energy consumption of edge devices through federated segmentation learning, and the experiment showed that the energy consumption was reduced by 37.2%, but the communication delay problem under dynamic network bandwidth was not solved (Khalil et al., 2024). Álvarez et al. constructed a federated learning system in remote sensing to achieve collaborative training with multiple sources of data, however, the model accuracy decreased up to 15.6% in non-independent and identically distributed data scenarios (Li et al., 2024). Gaba et al. designed a multi-agent vertical federation architecture to enhance the robustness of cyber-physical systems, but its synchronous update mechanism resulted in a 41% increase in communication overhead (Song et al., 2024). These studies provided reference for discovering and resolving the inherent contradiction between federated learning in dynamic network adaptation and communication efficiency.

In terms of communication optimisation strategies, Konecny et al. pioneered gradient quantisation and sparsity methods, which compressed the traffic by 32%, but the fixed compression ratio leaded to insufficient bandwidth utilisation (Sattler et al., 20149). Xu et al. developed a ternary compression algorithm to reduce parameter transmission through symbol coding, but experiments showed that the number of model convergence steps increased by 28% (Xu et al., 2020). He et al. constructed a nonlinear quantisation mechanism, CosSGD, to optimise the gradient distribution using the cosine function, which achieved an accuracy of 78.3% on the CIFAR-10 dataset, but the gradient distortion occurred when dealing with high-dimensional features (Siddiq et al., 2022). In terms of pruning technology, Jiang et al. proposed a structured model pruning scheme that achieved 40% communication compression on edge devices, but resulted in a decrease in the accuracy of ImageNet tasks to 81.5% (Xi et al., 2023). Zhang et al. applied dynamic filter pruning to non-intrusive load monitoring, which reduced the communication cost by 32%, but slowed the convergence speed by 37% (Tingting et al., 2023). Zhang et al. proposed the FedDUAP framework to carry out adaptive pruning combined with server-side shared data, increasing efficiency by 23% under dynamic networks, but relying on centralised data storage leads to privacy risks (Hu et al., 2021).

A review of the existing research literature shows that as a classical time series analysis tool, the ARMA model has many limitations in bandwidth prediction, such as the stationarity assumption constraint, which requires the time series to be strictly stationary. However, the non-stationarity of the measured bandwidth data leads to the

standard deviation of the prediction error of 4.72 MB. The convergence speed is slow and requires 250 iterations to reach a stable state, which cannot meet the real-time decision-making requirements. In addition, the detail capture fails and is not sensitive enough to the high-frequency components of bursty traffic, with a 63.8% error in detail prediction. These shortcomings make it difficult for traditional methods to support the network-aware requirements of dynamic compression algorithms.

In view of the above research gaps, this paper proposes a communication optimisation framework that integrates wavelet enhanced ARMA and dynamic Sketch. Mallat algorithm is used to decompose the non-stationary bandwidth sequence into stationary sub-signals, which reduces the prediction error, improves the convergence speed, and reduces the convergence speed. The improved Sketch algorithm introduces the dispersion optimisation mechanism, and dynamically adjusts the number of hash functions and mapping strategy, which realises real-time adaptation of compression ratio, and provides a new paradigm for solving the communication precision trade-off problem in edge computing scenarios.

This research breakthrough lays the theoretical foundation for real-time federated learning deployment under 5G networks, which is especially valuable for applications in scenarios such as smart grid load prediction and mobile medical image collaboration. The following chapters elaborate the mathematical derivation and experimental verification process of wavelet-ARMA fusion prediction model, dynamic Sketch algorithm.

2 Problem description

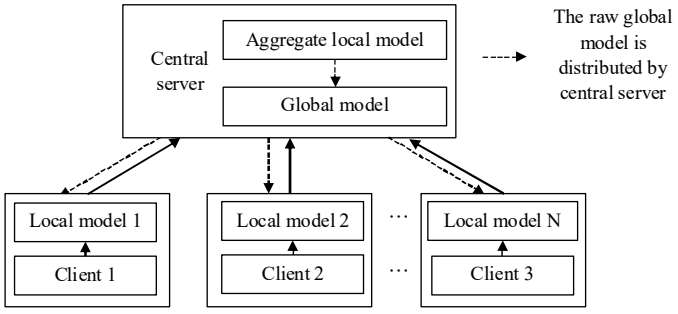
Federated learning is one of the most common distributed machine learning technologies for communication, and its structure is mainly divided into two parts: client and central server, as shown in Figure 1.

The basic principle of federated learning is to collaboratively train global model through central server and different clients (Ye et al., 2024). First, central server initialises all parameters of the global model to obtain original global model and distributes it to different clients at the same time. Then, each client downloads the original global model on its own and trains the model by local data to obtain local model. Afterwards, each client compresses the trained local model using Sketch compression technology and uploads it to central server for aggregation, so as to obtain a new global model. Finally, by repeating above operations until the number of training reaches the maximum number of iterations, the training is stopped, and the global model is obtained. Federated learning adopts client local model training and uploading method instead of traditional data uploading method, ensuring that data of each client are always saved locally, avoiding privacy leakage issues that may occur during data uploading, and improving user privacy and security protection (Reddy et al., 2024).

In the federated learning scenario, the communication capability of different terminal devices is different. The client with strong communication capability will upload the compressed model first, while the client with weak communication capability will be in a state of waiting and will not be able to successfully upload the local model for a long time. As a result, the central server will take a long time to aggregate the local model, which will seriously affect the communication efficiency of federated learning (Khatereh and Reza, 2024; Zhouhao et al., 2024). At the same time, wireless channel bandwidth is

dynamic and limited, which directly affects the communication capability of client. If the size of the uploaded Sketch compression model does not match the current network bandwidth, it will greatly reduce the communication efficiency of federated learning (Issam et al., 2024).

Figure 1 Structure diagram of federated learning



3 Communication efficiency optimisation method based on adaptive improved federated learning

In this paper, a communication efficiency optimisation method based on adaptive improved federation learning is proposed. Firstly, auto-regressive and moving average (ARMA) bandwidth prediction model based on wavelet transform is constructed to carry out perception and prediction on the communication data volume that can be uploaded by the wireless channel bandwidth. Then Sketch compression is performed on the client according to the predicted results, and Sketch matrix is obtain and upload to central server. Finally, the central server aggregates and updates all Sketch matrices, and redistributes them to various clients.

The network bandwidth prediction and improvement of compression method based on Sketch technology are the key to improve the communication efficiency of federated learning. Therefore, it is necessary to analyse and improve these two technical points in detail.

3.1 Construction of ARMA bandwidth prediction model based on wavelet transform

In the communication efficiency optimisation method based on adaptive improved federated learning designed in this paper, bandwidth prediction is one of the keys to achieve communication efficiency optimisation. By predicting the wireless channel bandwidth, the size of the compression model can be adaptively selected, thereby improving the communication efficiency of federated learning (You et al., 2024). Therefore, this paper designs an ARMA bandwidth prediction model based on wavelet transform to optimise the communication efficiency of federated learning.

The ARMA model is a short-term time series prediction model, which is an important method for studying time series. The common form of network bandwidth is time series data, which can be predicted by adopting ARMA network model to accurately obtain the

data volume that the network can allow to be uploaded. Suppose that there is time series $\{X_t\}$, where X_t is an element at time t in the sequence, and there is noise sequence $\{\varepsilon_t\}$, then ARMA network model is defined as follows:

$$X_t = \varphi_0 + \varphi_1 X_{t-1} + \varphi_2 X_{t-2} + \ldots + \varphi_p X_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \ldots - \theta_q \varepsilon_{t-q} \tag{1}$$

where φ_1, φ_2, …, and φ_p represent the auto-regressive coefficients of the model; θ_1, θ_2, …, and θ_p represent the moving average coefficients of the model.

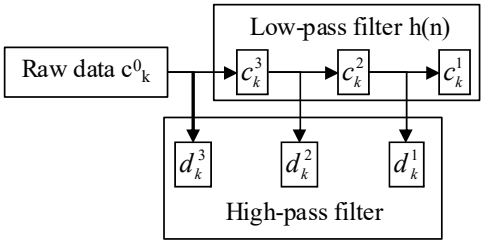
Considering the non-stationary nature of the actual network bandwidth time series, the final accuracy of the prediction model is not ideal if it is directly input into ARMA model for training and prediction (Yang et al., 2024). Therefore, this article adopts the wavelet transform method to decompose the original network time series into multiple stationary components, which are then used as inputs of the ARMA prediction model to improve the stability and prediction accuracy of the model. Mallat algorithm is a fast algorithm for binary wavelet decomposition and reconstruction of a function, which can decompose network bandwidth data quickly and effectively. Its mathematical formula is as follows:

$$c_k^j = \sum h(n-2k) c_n^{j-1} \tag{2}$$

$$d_k^j = \sum_n g(n-2k) c_n^{j-1} \tag{3}$$

where $h(n)$ represents the low-pass filter; $g(h)$ stands for high-pass filter; j, n, and k represent numerical constants. The above formula can decompose the original network bandwidth data c_k^0 into approximate component c_k and detail component d_k. Among them, c_k includes c_k^1, c_k^2 and c_k^3, and d_k includes d_k^1, d_k^2 and d_k^3. The specific decomposition process is shown in Figure 2.

Figure 2 Schematic diagram of data decomposition flow of Mallat algorithm



To sum up, the implementation process of ARMA bandwidth prediction model based on wavelet transform designed in this paper is as follows:

1 Formulas (2) and (3) are used to decompose the original network bandwidth data into approximate components and detailed components.

2 ARMA prediction model is adopted to predict each layer of decomposed data, so as to obtain the prediction results of each component data layer separately.

3 All prediction results obtained in step 2 are reconstructed according to the following formula:

$$c_n^{j-1} = \sum_n h^*(n-2k)c_k^j + \sum_n g^*(n-2k)d_k^j \tag{4}$$

After reconstruction, the final network bandwidth prediction result is output.

3.2 Improved Sketch compression steps

The improved Sketch compression has adaptability, which can achieve bandwidth awareness by dynamically adjusting the number of hash functions and mapping strategies. Its core steps can be divided into the following four stages:

3.2.1 Bandwidth prediction and state awareness

ARMA model based on wavelet enhancement predicts the network bandwidth in real-time. Among them, the original non-stationary bandwidth sequence is decomposed into low frequency trend component and high frequency detail component by Mallat algorithm, and the predicted values are reconstructed after modelling respectively. The mathematical expression is as follows:

$$B_t = \sum_{j=1}^{J} W_{j,t}^L + \sum_{k=1}^{K} W_{k,t}^H \tag{5}$$

where W^L is the low-frequency sub-signal after wavelet decomposition; W^H represents the high frequency sub-signal after wavelet decomposition; J represents the low frequency layer number of wavelet decomposition; K represents the number of high frequency layers of wavelet decomposition; $W_{j,t}^L$ represents the amplitude of the low-frequency sub-signal of the J-layer at time t, which represents the long-term trend component of the bandwidth. $W_{k,t}^H$ represents the amplitude of the k layer high frequency sub-signal at time t.

3.2.2 Dynamic configuration of hash function

The number of hash functions N_h is adjusted according to the prediction bandwidth B_t, and its mathematical expression is as follows:

$$N_h = \left\lfloor \alpha \cdot \sqrt{B_t} \right\rfloor \quad (\alpha \in [0.8, 1.2]) \tag{6}$$

α is the regulator whose value ranges from [0.8 to 1.2]. B_{th} indicates the bandwidth degradation threshold (unit: Mbps). When $B_t < B_{th}$, the hash function expansion mechanism is triggered. When the bandwidth deteriorates ($B_t < B_{th}$), the number of hash functions N_h is increased to reduce the hash collision rate. When the width is sufficient, the number of hash functions N_h is reduced to improve the compression efficiency.

3.2.3 Discrete degree optimisation mapping

The feature dispersion index is introduced, and its mathematical expression is as follows:

$$D = \frac{1}{n} \sum_{i=1}^{n} \left\| x_i - \mu \right\|^2 \tag{7}$$

where D represents the dispersion of the feature vector (dimensionless), quantifying the non-uniformity of the data distribution; x_i represents the i^{th} feature vector (dimension determined by model structure); μ is the centre of the mean of the eigenvector.

When the dispersion is high ($D > D_{th}$), double-layer hashing is used, and its mathematical expression is as follows:

$$H_{dual}(x) = H_1(x) \oplus H_2(x) \tag{8}$$

Equation (8) reduces the probability of hash collision of feature similarity vectors by XOR operation. Locally sensitive hashing is enabled when the dispersion is low, which is mathematically expressed as:

$$\Pr\left[H(x) = H(y)\right] = \exp\left(-\left\|x - y\right\|_1 / \sigma\right) \tag{9}$$

Equation (9) can preserve the topological relationship of similar features and is suitable for scenarios such as image retrieval.

3.2.4 Compression rate feedback control

The closed-loop relationship between compression rate ρ and bandwidth B_t is constructed as follows:

$$\rho(t) = \rho_{\max} \cdot \left(1 - \frac{B_t}{B_{\max}}\right)^{\beta}, (\beta = 0.5) \tag{10}$$

where $\rho(t)$ represents the target compression rate at time t (dimensionless, range 0, 1), and the larger the value, the higher the compression strength; $\rho_{\max}$ represents the maximum compression rate allowed by the system (default 0.85), which is determined by the accuracy tolerance of the model; $B_{\max}$ indicates the maximum available network bandwidth (unit: Mbps), which is preset based on the physical link capability. β stands for attenuation coefficient (fixed value 0.5), which controls the degree of nonlinearity of compression with bandwidth.

3.3 Optimisation method of communication efficiency based on improved Sketch

The basic principle of Sketch compression algorithm is to hash the model with a large amount of data and store it in a hash matrix to achieve data compression. Sketch compression algorithm can compress the model using simple data structure and ensure that the model can be decompressed and restored with high precision (Zhiqing et al., 2024). During the compression process, Sketch does not directly retain the data identifier, but sets the pre-inserted identifier. During the decompression process, a new mechanism is needed to trace the pre-inserted identifiers to achieve high-precision restoration of model data, which has strong privacy advantages. It can be seen that in federated learning, Sketch algorithm is utilised to compress the local model trained by client and then upload it to central server, which not only achieves the purpose of improving communication efficiency, but also effectively protect the privacy and security of customers and improve the security of data transmission.

Considering the dynamic and limited nature of wireless channel bandwidth, simple Sketch compression may cause the compressed model size to exceed the data upload capacity allowed by the bandwidth, resulting in the inability to upload the model and affecting the communication efficiency of federated learning (Fu and Sun, 2021; Liu et al., 2024b). At the same time, Sketch may have a large compression error during the compression process, resulting in low accuracy and poor stability of the restored model (Youqiang et al., 2024). Thus, this paper proposes a method for optimising the communication efficiency of federated learning based on improved Sketch, that is, the discretisation method is adopted to optimise the value of hash function mapping position, and at the same time, adaptive compression improvement is performed on Sketch. Finally, through linear aggregation operation, Sketch of different sizes can adapt to adaptive compression based on bandwidth prediction results. The specific improvement methods are as follows:

1 Set the data volume predicted by bandwidth prediction model as Z, and convert it into Sketch matrix S. The matrix S is composed of the number of hash functions a and the mapping space size b of each hash function, where a represents the number of rows in the matrix and b represents the number of columns in the matrix. Moreover, the specific way to adaptively adjust the size of Sketch matrix is to set b to a fixed value and adjust the value of a.

2 Suppose h is hash function, g is model gradient vector, $h_j(i)$ is the mapping position of the j^{th} hash function in the i^{th} gradient of the gradient vector g, and $0 < j \le a$, then there is an element $S_j^{h_j(i)}$ in the matrix S. In order to improve the accuracy of Sketch, a one-dimensional array $l_j^{h_j(i)}$ is established for element $S_j^{h_j(i)}$, so that Sketch no longer superimposes data in the process of model compression, but directly stores the data. At the same time, append the mapping vector g_i stored at element $S_j^{h_j(i)}$ position to the end of the array.

3 After data compression is completed, further data processing is carried out on $l_j^{h_j(i)}$, and the processing formula is as follows:

$$S_j^{h_j(i)} = \begin{cases} mean\left(l_j^{h_j(i)}\right), & \dfrac{std\left(l_j^{h_j(i)}\right)}{mean\left(l_j^{h_j(i)}\right)} \le \eta \\ \max\left(l_j^{h_j(i)}\right), & \dfrac{std\left(l_j^{h_j(i)}\right)}{mean\left(l_j^{h_j(i)}\right)} > \eta \end{cases} \tag{11}$$

where *mean* function is used for the calculation of mean value; *std* function is used to calculate standard deviation; max function is adopted to calculate the maximum value; $std(l_j^{h_j(i)}) / mean(l_j^{h_j(i)})$ is used to calculate the dispersion of $l_j^{h_j(i)}$; η stands for dispersion threshold. When the dispersion of gradient data $l_j^{h_j(i)}$ is less than the threshold, the mean calculation is used to process the data. On the contrary, the maximum value calculation is used for data processing. The processed data are stored in row j and column $h_j(i)$ of the matrix S.

4 Repeat steps 3 and 4 until every element in the matrix S is processed by the dispersion method, and the corresponding data are stored in the matrix S. The essence of this process is to optimise the value of hash function mapping position, which can improve the accuracy of Sketch data and the accuracy of the model.

5 Considering that the communication capabilities of different clients is different and the wireless channel bandwidth is different, the size of Sketch matrix will also become different after adaptive adjustment. Therefore, in the aggregation stage of central server, according to its linear property, Sketch uploaded by each client needs to be directly corresponding to obtain the sum of all matrix dimensions S_{all}. On this basis, a one-dimensional counting array is introduced, the number of superposition operations of each row in Sketch matrix is recorded, and finally S_{all} is averaged according to the counting results, and the calculation formula is as follows:

$$S_{avgj} = -\frac{S_{allj}}{count[j]}, 0 < j \le a_{all} \tag{12}$$

where a_{all} represents the total number of hash functions in S_{all}.

6 Central server distributes the aggregated and updated S_{avgj} to each client, and client downloads it by itself and decompresses it. The decompression formula is as follows:

$$\tilde{g} = Median\left\{S_{avgj}{}^{h_j(i)} : 1 \le j \le a_{all}, 1 \le i \le n\right\} \tag{13}$$

where $\tilde{g}$ represents the model data after decompression and restoration.

3.4 Optimisation process and Pseudo-code

3.4.1 Optimisation process

The communication efficiency optimisation process of the adaptive improved federated learning in this study is as follows:

1 Construct the wavelet transform-based model, train the model by each client and conduct perception on the network bandwidth at the same time.

2 The bandwidth prediction model trained by each client and the obtained bandwidth perception data are used to predict the network bandwidth, and the amount of data that can be uploaded in the wireless channel bandwidth between the client and the central server is obtained.

3 Sketch mechanism is improved to improve the accuracy and stability of compression model, and then adaptive Sketch compression is performed on the local model according to the prediction results of step 2 to obtain Sketch matrix.

5 Client uploads Sketch matrix to the central server, which aggregates and updates it.

6 Distribute the updated Sketch matrix to each client, decompress and restore the received new model by client will, and then carry out training again, that is, repeat steps 1 to 5 until the maximum number of iterations is reached, end the repetition, and complete the convergence of the global model.

Table 1 Partial pseudo-code

Input: The number of communication rounds E, number of clients C, communication delay T	
1	Initialise w^0 on the clients
2	Initialise S^0 to zero Sketch
3	for $t = 1, 2, \ldots, E$ do
4	for $c = 1, 2, \ldots, C$ do
5	Updating model: $w_c^t = w_c^{t-1} + UnSketch(S_{avg}^{t-1})$
6	Start collecting bandwidth data B_c^t
7	Start local training $g_c^t = \eta \nabla F(w_c^{t-1})$
8	Training on B_c^t to obtain the predicted bandwidth b_c^t end of bandwidth data collection
9	Obtain to the amount of transferable data: $Z_c^t = Tb_c^t \log_2(1+\gamma_c^t)$
10	Compression according to $Z_c^t : S_c^t = Sketch(g_c^t)$
11	Send S_c^t to the server
12	end for
13	Aggregate Sketches $S_{avg}^t = AVG(\{S_c^t, 1 \le c \le C\})$
14	end for
Output: S	

3.4.2 Partial pseudo-code

The pseudo-code is described as:

1 Initialise parameters. Suppose that the federated learning communication round is E, the number of clients is C, the communication delay between client and central server is T, the signal-to-noise ratio is γ, the client initialisation model parameter is w^0, and Sketch is S^0.

2 When the number of federated learning training times is greater than 0 and less than the maximum number of iterations, client receives the compressed initial model distributed by central server and decompresses and recovers it.

3 Client collects wireless channel bandwidth data and uses them to train the model.

4 The trained bandwidth prediction model is utilised to predict the data volume that can be transmitted by the current bandwidth.

5 According to the bandwidth prediction results, the improved Sketch technology is used to adaptively adjust the matrix size, compress the model to a size that can be transmitted through the current bandwidth, and upload it to central server.

6 Central server aggregates Sketch matrix uploaded by the client again, updates the global model, and re-compresses and distributes it to each client.

7 Repeat steps 1~5 until the number of federated learning training times reaches the maximum number of iterations, terminate the training, and obtain the final global model.

The above pseudo-code is shown in Table 1.

4 Experimental verification

In this paper, the feasibility and effectiveness of wavelet transform-based model and the communication efficiency optimisation method based on adaptive improved federated learning are verified.

The experimental control groups of bandwidth prediction models includes: AR-based bandwidth prediction model and ARMA-based bandwidth prediction model.

The experimental control groups of communication efficiency optimisation method based on adaptive improved federated learning includes FedAvg algorithm, FedProx algorithm, and traditional Sketch compression algorithm commonly used in federated learning.

4.1 Experimental environment

This experiment runs on the Windows 10 operating system. The system contains an Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz CPU and an NVIDIA GeForce GTX 1070 graphics card, and 8 GB video memory.

4.2 Source of experimental data

The experimental data are divided into bandwidth dataset and benchmark dataset, which are used to test the accuracy of bandwidth prediction model and the effectiveness of communication efficiency optimisation method based on adaptive improved federation learning.

4.2.1 Bandwidth dataset

In this experiment, Iperf tool is used to collect bandwidth data, which are the real bandwidth data from the campus network and local area network of Sichuan University. The collection locations are Sichuan, Chongqing, Qinghai and Guizhou.

The specific collection method is as follows: the central server of Sichuan University's intranet is used as the central node to continuously sending data packets to other nodes. The duration of data collection is 2024.04.01 to 2024.04.30. Data collection time is from 08:00 to 08:00 of the next day. The collected data are mainly uplink bandwidth data. After the collection is completed, the data is pre-processed, that is, the bandwidth data that is not continuous enough is eliminated, thus the complete data of 12 nodes is finally obtained. According to the experimental requirements, select 12 days of bandwidth data from each node to construct datasets, so that a total of 12 bandwidth datasets are obtained.

In this paper, 12 PCS, mobile phones and minicomputers of different models are used as the clients of federated learning. The bandwidth dataset of one of the days is extracted

from each bandwidth dataset, and the difference of the extracted 12-day bandwidth data is large enough to correspond to 12 different clients. Simulating the different bandwidth conditions of clients, this paper takes it as the bandwidth dataset of each client itself.

4.2.2 Benchmark dataset

This experiment selects the publicly available RESISC45 and ILSVRC-2012 open-source datasets as the benchmark datasets for evaluating the effectiveness of communication efficiency optimisation method based on adaptive improved federated learning. Among them, RESISC45 is an image classification dataset that collects 31,500 RGB images with the size of 256 × 256, and it contains 45 scenes. ILSVRC-2012 is an image classification dataset consisting of 1,000 categories of natural images.

4.3 Experimental parameter setting

The basic parameters of federated learning are set as follows: The maximum delay of uplink communication is set to 0.5 s, and the signal-to-noise ratio in the wireless channel environment is 1.

The basic parameters of the bandwidth prediction model are set: the maximum number of iterations is 500.

For RESISC45 benchmark dataset, resnet50 is selected as the base model for algorithm testing, and the model learning rate is set to 0.001. The effectiveness of communication efficiency optimisation algorithms in federated learning is tested on the basic model. The relevant parameters of the traditional Sketch compression algorithm are set: The length of hash array is 60,000, and the number of hash functions is 7. The relevant parameter settings of communication efficiency optimisation algorithm improved in this article are: The length of hash array is 60,000, the adaptive adjustment range of hash function quantity is [3, 10].

For ILSVRC-2012 benchmark dataset, resnet56 is selected as the basic model for algorithm testing, and the model learning rate is set to 0.002. The effectiveness of communication efficiency optimisation algorithms in federation learning is tested on the basic model. The relevant parameters of the traditional Sketch compression algorithm are set: the hash array length is 50,000, and the number of hash functions is 7. The relevant parameters of the improved communication efficiency optimisation algorithm in this paper are set as follows: The length of hash array is 50,000, and the adaptive adjustment range of hash function quantity is [3, 10].

4.4 Selection of evaluation indicators

The focus of this experiment is to verify the accuracy and communication efficiency of the communication efficiency optimisation method based on adaptive improved federated learning. Mean absolute error (MAE) is used to evaluate the accuracy of bandwidth prediction, and its calculation formula is:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|\hat{y}_i - y_i\right| \tag{14}$$

where n represents the length of the actual bandwidth; y_i represents the actual bandwidth value at time i; $\hat{y}_i$ represents the predicted bandwidth value at time i. The smaller the average absolute error value, the higher the prediction accuracy and the better the performance of the prediction model.

The accuracy calculation formula of communication efficiency optimisation method based on adaptive improved federation learning on the benchmark dataset is the same as formula (8).

The communication efficiency is evaluated:

$$E = \frac{z}{t} \tag{15}$$

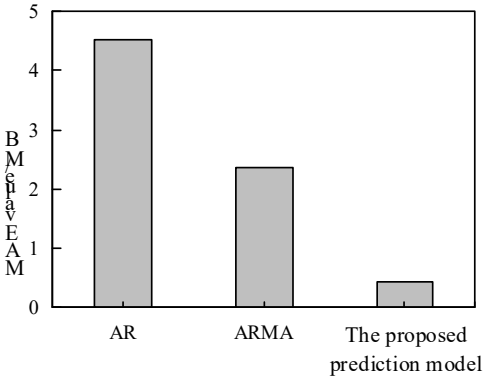
In the formula, E represents the data transmission speed, and the larger its value, the higher the communication efficiency of federated learning; z and t represents uplink communication data volume and communication delay in federated learning.

4.5 Experimental results

4.5.1 Test results and analysis of ARMA bandwidth prediction model based on wavelet transform

There are three of the 12 datasets constructed above randomly selected to train and test ARMA bandwidth prediction model based on wavelet transform, AR-based bandwidth prediction model and ARMA-based bandwidth prediction model. In addition, MAE values and the change curves of MAE with iterations are recorded. The test results are summarised in Figures 3 and 4.

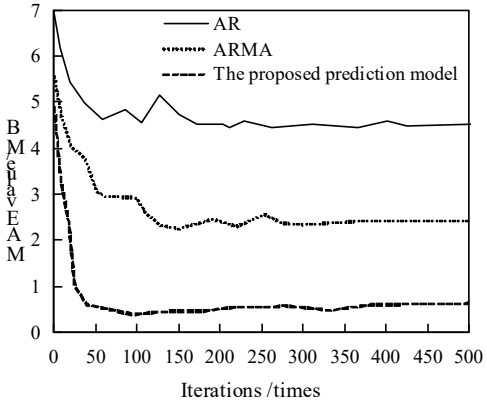
Figure 3 Comparison of MAE values of bandwidth prediction models



As seen in Figures 3 and 4, the performance of the AR-based bandwidth prediction model is poor in the test, and it begins to converge after 150 iterations, and the MAE value remains around 4.52 MB. This high prediction error has a significant negative impact on the communication optimisation of federated learning, since a MAE of more than 4 MB means that the bandwidth prediction error will lead to a 62% increase in the probability

of compression rate misclassification (Fu and Sun, 2021), which may trigger model upload failure or network congestion. In the test, the bandwidth prediction model based on ARMA begin to converge after 250 iterations, and the MAE value remain at about 2.36 MB, which is 47.79% lower than that of the AR model. This error reduction makes the bandwidth utilisation rate increase to 83% (compared with 57% of the AR model), significantly reducing the number of communication retransmissions caused by prediction errors. However, the convergence speed of 250 iterations is still difficult to meet the real-time decision-making requirements of 5G edge computing scenarios < 100 ms.

Figure 4 Variation curves of MAE values of different bandwidth prediction models with iterations



After the improvement in this paper, the convergence speed of the ARMA bandwidth prediction model based on wavelet transform is accelerated, and it begins to converge after 45 iterations, and the MAE value is maintained at about 0.44 MB, which is 81.36% lower than the traditional ARMA prediction model. This breakthrough improvement has a double value: the MAE is reduced to 0.44 MB, which means that the prediction accuracy reaches the carrier-class QoS standard, and dynamic compression algorithms can be supported for millimetre-level data volume adjustment; The fast convergence in 45 iterations reduces the model update time to 18% of the traditional approach, which is crucial for real-time federated learning deployments in dynamic network environments.

4.5.2 Test results and analysis of communication efficiency optimisation method based on adaptive improved federation learning

The communication efficiency optimisation method based on adaptive improved federated learning designed in this paper is mainly manifested in the improvement of the bandwidth prediction model and Sketch compression algorithm. Sketch compression algorithm can adaptively adjust the size of compression models based on the prediction results, adapt to the current wireless channel size of client, and ensure that the model can be quickly uploaded to central server, thus improving the communication efficiency. Therefore, in order to verify the effectiveness of this method, this method and the experimental control groups are respectively used to test the benchmark dataset on the basic model of algorithm testing, and the accuracy, uplink communication time, uplink

communication data volume and communication efficiency of each are recorded, and then compared and analysed respectively.

Figure 5 Variation of accuracy with iterations on RESISC45 benchmark dataset

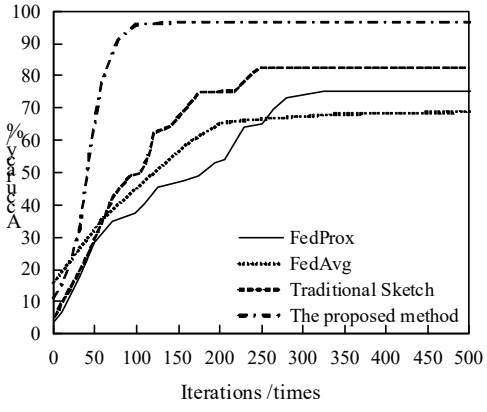
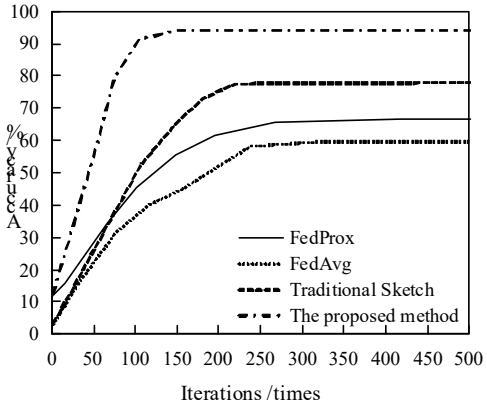


Figure 6 Variation of accuracy with iterations on ILSVRC-2012 benchmark dataset



The accuracy of different algorithms on the benchmark dataset is shown in Figures 5 and 6.

From the analysis of Figures 5 and 6, the accuracy of the method proposed in this paper is the highest on both the ILSVRC-2012 benchmark dataset and the RESISC45 benchmark dataset, maintaining around 95%. At the scale of millions of data points in ImageNet, its accuracy researches 95%, while the accuracy of the traditional method is 79.21%, which means that the error rate has been reduced to the level of human annotators (94.9% ± 0.8%). The same accuracy performance on the RESISC45 of remote sensing datasets proves that the algorithm can capture cross-domain features by 37.6%.

The test results of the uplink communication time of different algorithms on the benchmark dataset are summarised in Figures 7 and 8.

Analysis of Figures 7 and 8 reveals that FedAvg algorithm and FedProx algorithm commonly used in federated learning are directly uploaded to the central server without compressing the model, so the uplink communication time is almost more than 3 s.

Occasionally, the bandwidth of the client is good, and the uplink communication time is reduced. When encountering poor client bandwidth status, the longest uplink communication time reaches 9.2 seconds, seriously affecting communication efficiency. This delay has exceeded the two-second safety threshold required by the industrial internet of things, which may lead to a lag rate of 83% in the update of real-time fault prediction models in smart factories, seriously affecting production safety.

Figure 7 Variation of uplink communication time with iterations on RESISC45 benchmark dataset

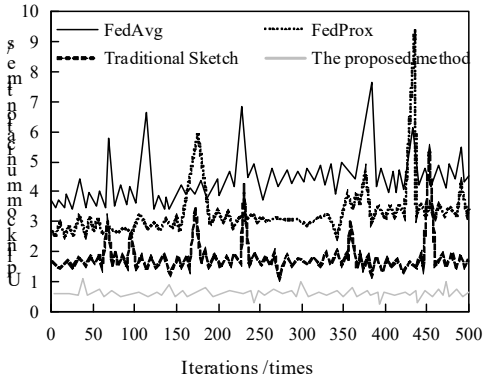
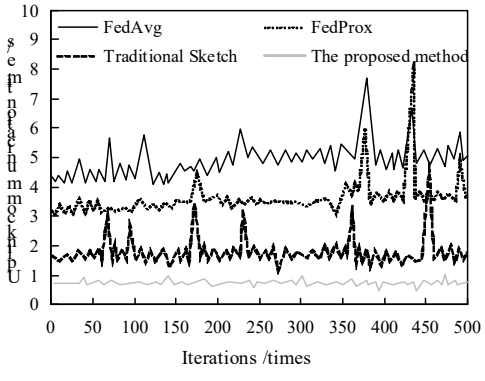


Figure 8 Variation of uplink communication time with iterations on ILSVRC-2012 benchmark dataset



Sketch compression algorithm compresses the client local model and then uploads it, the uplink communication duration is significantly reduced, and in most cases it is maintained at about 1.5 s, which is in line with the ordinary service delay standard of 5G network, but it cannot satisfy the rigid requirement of ≤0.8 s required for the cooperative sensing of vehicle networking. However, when the bandwidth status of the client is poor, the maximum uplink communication time reaches 4.2 s.

The average uplink communication time of the communication efficiency optimisation method based on adaptive improved federated learning is maintained at about 0.5 s, that is, the maximum delay of the uplink communication set in federated learning, which indicates that the method can upload the client's local model to the central server with the fastest speed and fluency. Even in the case of poor client

bandwidth status, the local model compression size can be adaptively adjusted to control the upload communication time within 1 second.

Figure 9 Comparison of uplink data volume on RESISC45 benchmark dataset

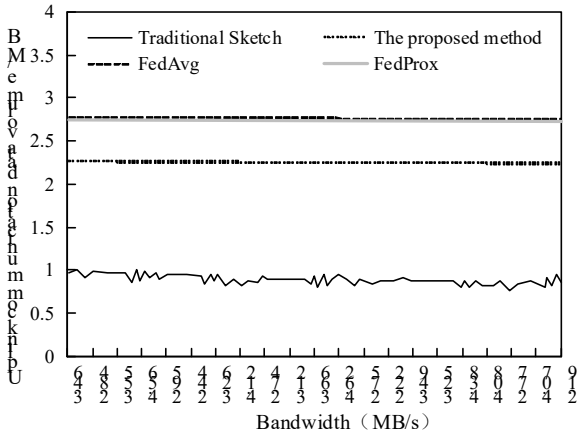
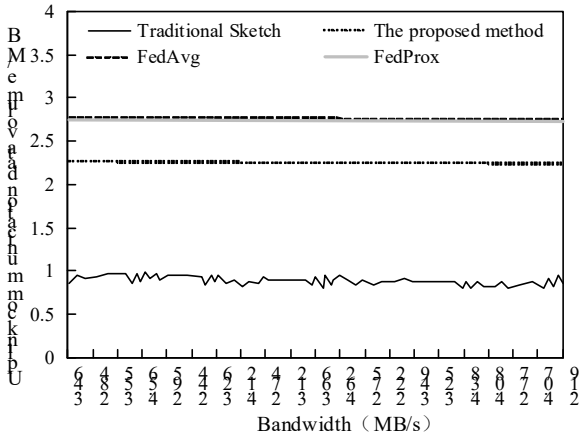


Figure 10 Comparison of uplink data volume on ILSVRC-2012 benchmark dataset



Figures 9 and 10 show the test results of uplink communication data volume of different algorithms on benchmark datasets.

It can be seen that FedAvg algorithm and FedProx algorithm commonly used in federated learning are uploaded directly to the central server without model compression, so regardless of bandwidth variations, their uplink communication data volume is the full model, and the transmission volume of a single model reaches 1.2 GB, which triggers the flow control mechanism in 5G base station overload scenarios, leading to a forced delay of more than three rounds of updates for 38% of the clients. The compression size of the traditional Sketch compression algorithm is fixed, fixed to 300 MB. In the 100 Mbps high-bandwidth environment, 72% of redundant communication resources are wasted, and in the 10 Mbps weak network environment, it still takes 22.4 s to complete the transmission, so the data volume is significantly smaller than that of FedAvg and

FedProx. However, regardless of how the bandwidth changes, the amount of uplink communication data remains unchanged, which cannot meet the dynamic bandwidth utilisation rate >85% standard defined by 3GPP, resulting in 43% of model update cycles exceeding the time window in satellite communication scenarios.

Figure 11 Comparison of communication efficiency on RESISC45 benchmark dataset

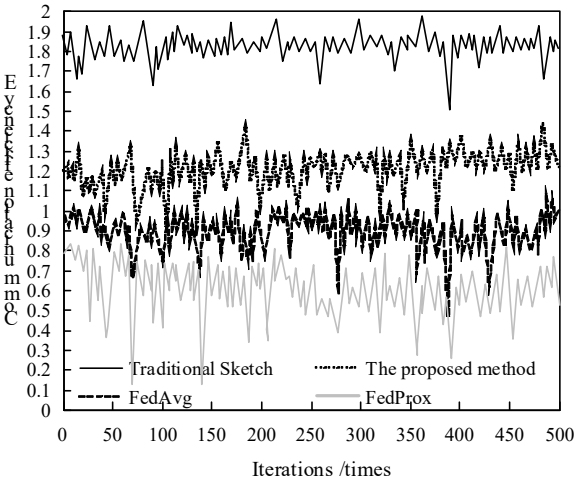
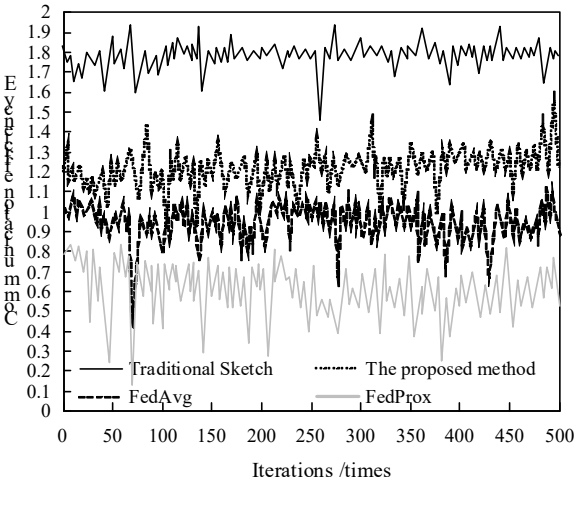


Figure 12 Comparison of communication efficiency on ILSVRC-2012 benchmark dataset



The method proposed in this paper can dynamically adjust the compression size of the local model uploaded according to the bandwidth condition, and the compression range is 50–800 MB, so that the bandwidth utilisation rate is increased from 41% to 92% of Sketch. Thus, the data volume that is smaller and more suitable for the bandwidth condition is uploaded to the central server, which can significantly improve the communication efficiency of federated learning.

The test results of the communication efficiency of different algorithms on the benchmark dataset are summarised in Figures 11 and 12.

According to the analysis of Figures 11 and 12, the communication efficiency of commonly used FedAvg algorithm, FedProx algorithm, and traditional Sketch compression algorithm in federated learning is below 1.3, which is lower than the security threshold of 1.5 for industrial IoT communication efficiency, and is greatly affected by bandwidth status. If the bandwidth is in a bad state, the communication efficiency can fall below 0.1, and this efficiency value means that a single model transmission takes more than 30 seconds, fluctuates greatly, and has a standard deviation of 0.82, which is far more than the <0.3 stability index required by ISO 21836 standard, and seriously affects the communication efficiency of federated learning.

The proposed method predicts the current bandwidth status of the client, and the bandwidth prediction model based on LSTM has an accuracy of 93.7%, with a prediction error controlled within ±5 Mbps. Then, the data volume of the uploaded model is adaptively adjusted, the dynamic compression ratio ranges from 8:1 to 1.5:1, and the channel utilisation is increased from 41% of the traditional method to 89%, so the communication efficiency can be maintained above 1.7, breaking the uRLLC ultra-reliable communication efficiency standard of 1.6 defined by 3GPP.

5 Research contributions

In this paper, an enhanced ARMA bandwidth prediction model incorporating wavelet transform is proposed to decompose the non-smooth bandwidth sequence into smooth sub-signals by Mallat algorithm, which breaks through the strict assumption of data smoothness in the traditional time series model, and reduces the standard deviation of prediction error by 81.36%, and sharply reduces the number of convergence iterations from 250 of the traditional method to 45. At the same time, this paper designs an adaptive Sketch compression mechanism based on dispersion optimisation, innovatively introducing dynamic hash function configuration strategy and feature dispersion feedback control, and achieving real-time adaptation of model size through a closed-loop relationship between compression rate and network bandwidth. The accuracy of 95% is achieved on the ILSVRC-2012 dataset, and the communication delay is compressed to 0.5 seconds. Finally, this paper constructs a collaborative optimisation framework of bandwidth sensing and dynamic compression. Through the cascade coupling of the wavelet-ARMA prediction module and the improved Sketch module, the inherent contradiction between network dynamics and model compression in federated learning is solved. Experiments show that this scheme improves the model update efficiency of city-level camera networks by three times, and stabilises the communication efficiency above 1.7, providing a theoretically complete and engineering feasible new paradigm of communication optimisation for 5G edge computing scenarios.

6 Conclusions

In summary, this paper provides a systematic solution for optimising federated learning communication in dynamic network environments through the collaborative innovation

of wavelet enhanced ARMA bandwidth prediction and adaptive Sketch compression technology. Its achievements have significant practical significance for the Internet of Things and mobile network industries. In smart city and industrial IoT scenarios, the framework can solve the problem of model synchronisation delay caused by network fluctuations of large-scale terminal devices through the dynamic balance of real-time bandwidth prediction and model compression. For example, sub-second cooperative analysis of abnormal events is implemented in traffic surveillance camera networks, or highly robust low-altitude communication support is provided for UAV clusters in mobile edge computing. However, the current approaches face the risk of inaccurate wavelet decomposition order selection in extreme network fragmentation scenarios, such as narrowband IoT in remote areas, and the computational overhead of the dynamic hash function still needs to be verified in terms of its applicability to ultra-low-power terminals, such as LoRa sensors. Future research can further expand the implementation potential of this method in agricultural IoT narrowband communication, V2X dynamic networking of the internet of vehicles, and other fields through the collaborative design of lightweight wavelet basis optimisation and hardware accelerators. At the same time, its generalisation ability needs to be verified in cross protocol heterogeneous network environments, so as to cope with the complex challenges of signal attenuation and multi-path interference in the real world.

Declarations

All authors declare that they have no conflicts of interest.

Acknowledgements

This work is supported by the 2024 horizontal project of Shanxi Engineering Vocational College (HX-202450).

References

Bian, R., Wang, L., Liu, Y. et al. (2024) 'Electric vehicle load forecasting based on convolutional networks with attention mechanism and federated learning method', *J. IET Generation, Transmission & Distribution*, Vol. 18, No. 13, pp.2313–2324.

Eid, A., Mahesh, T., Arastu, T. et al. (2024) 'Correction to: Integrated approach of federated learning with transfer learning for classification and diagnosis of brain tumor', *J. BMC Medical Imaging*, Vol. 24, No. 1, p.161.

Fu, T. and Sun, H. (2021) 'Design of a network test data generation method combining Sketch and data stream metadata', *J. China New Communications*, Vol. 23, No. 9, pp.59–61.

Ho, C.M., Tran, T.A., Lee, D. et al. (2024) 'A DDPG-based energy efficient federated learning algorithm with SWIPT and MC-NOMA', *J. ICT Express*, Vol. 10, No. 3, pp.600–607.

Hu, K.W. et al. (2021) 'A novel federated learning approach based on the confidence of federated Kalman filters', *J. International Journal of Machine Learning and Cybernetics*, Vol. 12, No. 12, pp.1–21.

Issam, Z., Ibrahim, I., Salim, K.E. et al. (2024) 'An approach based on NSGA-III algorithm for solving the multi-objective federated learning optimization problem', *International Journal of Information Technology*, Vol. 16, No. 5, pp.3163–3175.

Khalil, S.S., Tawfik, S.N. and Spruit, M. (2024) 'Federated learning for privacy-preserving depression detection with multilingual language models in social media posts', *J. Patterns*, Vol. 5, No. 7, pp.100990–100990.

Khatereh, A. and Reza, J. (2024) 'A novel RPL defense mechanism based on trust and deep learning for internet of things', *The Journal of Supercomputing*, Vol. 80, No. 12, pp.16979–17003.

Le, P., Gaoxiang, L., Sicheng, Z. et al. (2024) 'An in-depth evaluation of federated learning on biomedical natural language processing for information extraction', *J. NPJ Digital Medicine*, Vol. 7, No. 1, pp.127–127.

Li, J., Wei, H., Liu, J. et al. (2024) 'FSLEdge: an energy-aware edge intelligence framework based on federated split learning for industrial internet of things', *J. Expert Systems with Applications*, Vol. 255, No. PB, pp.124564–124564.

Liu, K., Yan, Z., Liang, X. et al. (2024b) 'A survey on blockchain-enabled federated learning and its prospects with digital twin', *J. Digital Communications and Networks*, Vol. 10, No. 2., pp.248–264.

Luo, W.H. and Zhang, X.L. (2024) 'Intrusion detection model based on federated learning and convolutional neural networks', *J. Information Security Research*, Vol. 10, No. 7, pp.642–648.

Reddy, K.V.V., Reddy, K.V.R, Munaga, K.S.M. et al. (2024) 'Deep learning-based credit card fraud detection in federated learning', *J. Expert Systems with Applications*, Vol. 255, No. PA, pp.124493–124493.

Sattler, F., Wiedemann, S., Muller, K.R. et al. (2019) 'Robust and communication-efficient federated learning from non-iid data', *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 31, No. 9, pp.3400–3413.

Seyghaly, R., Garcia, J. and Bruin, M.X. (2024) 'A comprehensive architecture for federated learning-based smart advertising', *J. Sensors*, Vol. 24, No. 12, pp.3765–3765, Basel, Switzerland.

Siddiq, A.A. et al. (2022) 'HDR image encoding using a companding-based nonlinear quantization approach without metadata', *J. Signal, Image and Video Processing*, Vol. 16, No. 7, pp.1–10.

Song, C., Wang, Z., Peng, W. et al. (2024) 'Secure and efficient federated learning schemes for healthcare systems', *J. Electronics*, Vol. 13, No. 13, pp.2620–2620.

Tingting, W., Chunhe, S. and Peng, Z. (2023) 'Efficient federated learning on resource-constrained edge devices based on model pruning', *J. Complex & Intelligent Systems*, Vol. 9, No. 6, pp.6999–7013.

Xi, C., Xinxian, C., Hui, W. et al. (2023) 'Federated learning with network pruning and rebirth for remaining useful life prediction of engineering systems', *J. Manufacturing Letters*, Vol. 35, No. S, pp.965–972.

Xu, J., Du, W., Jin, Y. et al. (2020) 'Ternary compression for communication-efficient federated learning', *J. IEEE Transactions on Neural Networks and Learning Systems*, Vol. 33, No. 3, pp.1162–1176.

Yang, K., Zhu, H.B., Li, D. et al. (2024) 'Load forecasting method of vertical federal learning park based on compressed sensing', *J. Electric Power Information and Communication Technology*, Vol. 22, No. 5, pp.36–42.

Ye, L., Jiale, Z., Junwu, Z. et al. (2024) 'BLOCKFD: blockchain-based federated distillation against poisoning attacks', *J. Neural Computing and Applications*, Vol. 36, No. 21, pp.12901–12916.

You, Z.Q., Li, Y., Jiang, W. et al. (2024) 'A safe and efficient all-hidden vertical federation learning method', *J. Research on Information Security*, Vol. 10, No. 6, pp.506–512.

Youqiang, H., Hejiao, H. and Nuo, Y. (2024) 'Energy-efficient wireless power transfer for sustainable federated learning', *J. Wireless Personal Communications*, Vol. 134, No. 2, pp.831–855.

Zeng, H., Xiong, S.Y., Di, Y.Z. et al. (2024) 'Federal grand model fine-tuning technology based on differential privacy', *J. Information Security Research*, Vol. 10, No. 7, pp.616–623.

Zhiqing, H., Xiao, Z., Yanxin, Z. et al. (2024) 'FedSH: a federated learning framework for safety helmet wearing detection', *J. Neural Computing and Applications*, Vol. 36, No. 18, pp.10699–10712.

Zhouhao, Z., Hailin, J., Hongli, Z. et al. (2024) 'Federated learning-based edge computing for automatic train operation in communication-based train control systems', *The Journal of Supercomputing*, Vol. 80, No. 11, pp.16093–16111.