Secure IoT Data Sharing via Semi-Supervised Clustering Federated Learning with Fuzzy Multi-Objective Decision-Making and SecureBoost Integration

Lihong Zhang[1]*, Kai Yan[1], Xia Yang[2]
[1]Department of Artificial Intelligence, Laiwu Vocational and Technical College, Ji'nan, 271100, China
[2]Technology Department, Ji'nan City Laiwu District Media Convergence Center, Ji'nan, 271100, China
E-mail: zhanglihong1129@outlook.com
*Corresponding author

Keywords: clustering federated learning, semi-supervised learning, data sharing, fuzzy multi-objective decision-making method, security federation ascension tree

Received: September 22, 2025

The Internet of Things enables digital transactions and data sharing, but poses significant security risks during data transmission. To address the issues of weak data sharing security and stability, this study proposes a data sharing technology for Internet of Things. The framework integrates semi-supervised clustering with fuzzy multi-objective decision making and SecureBoost encryption, evaluated on USPS and synthetic datasets. Experimental results show that the Semi-Supervised Clustering Federated Learning algorithm achieves 94.6% accuracy on synthetic datasets, outperforming Multi-View Deep Subspace Clustering Networks (90.3%) and Mid-level Deep Pattern Mining (84.6%). Furthermore, evaluation of the proposed fusion data sharing technology reveals that the key encryption time remains within 250 ms for files smaller than 10 MB. For a 1 MB file, the decryption time is 19 ms. These results demonstrate that the proposed technology prevents data leakage and enables secure multi-party transactions. This study contributes to future secure access to diverse resource data in Internet of Things and ensures fair data sharing.

Povzetek: Študija predstavlja varno tehnologijo za deljenje podatkov v internetu stvari, ki izboljšuje zaščito in učinkovitost prenosa podatkov.

1 Introduction

The Internet of Things (IoT) enables smart applications in urban transportation, healthcare, and other domains. It collects data from smart terminals through cloud servers for applications ranging from urban management to industrial automation [1]. However, IoT expansion has introduced data sharing security challenges, such as inefficient data service transactions and leakage of corporate core secrets [2]. To prevent illegal access or tampering of data during multi-party data sharing in IoT, a joint protection mechanism through cloud computing platforms and encryption technologies is often employed. This approach enables dynamic management while ensuring data confidentiality. Among them, the commonly used encryption protocols for IoT data transmission include Secure Sockets Layer and Transport Layer Security. These protocols can authenticate information through asymmetric and symmetric encryption technologies, protecting data from eavesdropping [3]. Although these technologies strengthen IoT data sharing security, they remain insufficient for collaborative privacy protection. Therefore, many experts and scholars have used algorithms to enhance data privacy security. Among them, the Vertical Federated Learning system is a commonly used framework for IoT data sharing. It allows multiple participants to collaborate in training simultaneously, avoiding personal privacy leakage [4-5]. However, this system still cannot prevent information attacks such as abnormal errors, posing significant threats to IoT data security. To address problems such as incorrect data flow and chaotic resource sharing in IoT, this study proposes an improved Semi-Supervised Learning Federated Learning (SL-FL) clustering algorithm. This algorithm is used to construct a fused IoT data sharing technology that provides real-time protection for IoT data resources. The technology also employs SecureBoost to perform homomorphic encryption on sensitive IoT data. It is expected that this technology will significantly enhance IoT data sharing and reduce the risk of malicious attacks on IoT aggregation results.

The study proposes a federated learning algorithm integrating SL for IoT data sharing and privacy protection. Addressing the challenges of complex data feature distributions and scarce labelled samples in IoT environments, the algorithm achieves improved clustering accuracy and stability while safeguarding local privacy. The study introduces a fuzzy multi-objective decision-making method to enhance the extraction and discrimination capabilities of multi-source heterogeneous data features, further improving clustering performance. To address security risks, the study designs a secure

framework integrating SecureBoost homomorphic encryption with blockchain technology to achieve traceability and tamper-proof data transmission, ensuring privacy security in multi-party interactions.

2 Related works

Securing IoT resource sharing against malicious attacks requires robust data sharing technologies. Vatambeti et al. proposed a blockchain-based data broadcasting strategy to address IoT medical data security issues. Experimental results demonstrated that this technology created a secure and reliable platform, enabling multiple data providers to exchange data and record it in a shared ledger [6]. Li et al. proposed a multi-teacher knowledge distillation framework to address the issues of representation bias and spatial inconsistency caused by the heterogeneity and extreme skewness of categories in IoT cross-device architectures. Through adaptive knowledge integration, weighted joint loss, and global anchoring alignment, they unified the representation space and improved local representation capabilities [7]. Patil et al. proposed a new attribute-based signcryption scheme to address concerns about privacy protection and access control of sensitive electronic health records in medical IoT. Experimental results indicated that this scheme supported multiple security features, such as data confidentiality and efficient access control authentication [8]. Wang et al. proposed an IoT and fuzzy logic-based crop pest control system to solve agricultural information management issues. This system, composed of IoT technology and mobile applications, provided real-time environmental information for farm management, and the experimental results proved its effectiveness [9].

As IoT data continues to be refined, related enterprises and designers have proposed higher requirements for IoT data aggregation results. They demand not only security during IoT data transmission but also the effective aggregation of transmission results across multiple domains. Therefore, researchers have also conducted related studies on clustering algorithms. Zubair et al. proposed an improved K-means clustering algorithm to reduce iteration counts and execution time in data modeling. The algorithm was evaluated using a synthetic dataset with 10 M instances across multiple dimensions. Experimental results showed that this method outperformed traditional K-means and random centroid initialization methods in terms of computation time and iteration counts [10]. Liu et al. proposed a node clustering algorithm based on node embedding for heterogeneous information networks to ensure adjacent features were not overlooked. Experimental results demonstrated that this algorithm effectively clustered heterogeneous information while fully considering the relationships among target objects [11]. Sinha et al. proposed two parameter-free particle swarm optimization algorithms to address the issue of requiring predefined cluster numbers in traditional clustering algorithms. These algorithms introduced a new version of mutant PSO. Experimental results showed that both algorithms outperformed classical algorithms across eight real-world datasets [12]. Abedpour et al. proposed a clustering fusion method combining genetic algorithms and K-means to improve IoT resource allocation efficiency. Experimental results indicated that the processing nodes selected at the fog layer helped minimize latency and allowed applications to access resources simultaneously [13]. Dey et al. proposed a new quantum-inspired differential evolution algorithm for automatic clustering in unlabeled data to enhance robustness. Experimental results demonstrated that this algorithm achieved good convergence speed and better computational results [14].

In summary, researchers worldwide have conducted related studies on IoT data and achieved significant progress in IoT data sharing security. However, there is still limited research on IoT shared data security and joint clustering. Therefore, this study develops an IoT data sharing technology based on SL and proposes an improved SL-FL algorithm to enhance IoT data privacy protection. Table 1 shows the comparative results of existing studies.

Table 1: Comparison of existing studies

Reference	Method	Dataset	Results	Limitations
Vatambeti et al. [6]	Blockchain-based data broadcasting strategy	Medical IoT data	Created a secure and reliable platform	Scalability issues for larger datasets and dynamic environments
Li et al. [7]	Multi-teacher knowledge distillation framework	IoT cross-device architecture	Improved local representation capabilities	Focuses mainly on cross-device heterogeneity issues
Patil et al. [8]	Attribute-based signcryption scheme	Medical IoT	Supports multiple security features	Limited scalability and effectiveness outside of medical IoT
Wang et al. [9]	IoT and fuzzy logic-based crop pest control system	Agricultural IoT	Effectively provided real-time environmental information for farm management	Lack of integration with broader IoT security mechanisms

Zubair et al. [10]	Improved K-means clustering algorithm	Synthetic dataset (10 million instances)	Significant improvements in computation time and iteration counts	Not tested on real IoT datasets
Liu et al. [11]	Node embedding-based clustering algorithm for heterogeneous information networks	Heterogeneous information networks	Effectively clustered heterogeneous information	Did not explicitly address data security during clustering
Sinha et al. [12]	Parameter-free particle swarm optimization clustering algorithm	Eight real-world datasets	Significant performance improvement	Requires predefined cluster numbers
Abedpour et al. [13]	Clustering fusion method based on genetic algorithms and K-means	IoT resource allocation data	Minimized latency	Poor fault tolerance
Dey et al. [14]	Quantum-inspired differential evolution automatic clustering algorithm	Unlabeled data	Improved convergence speed	Did not focus on data privacy issues
This paper	IoT data sharing technology based on semi-supervised clustering, fuzzy multi-objective decision-making, and SecureBoost encryption	Synthetic dataset, USPS dataset, real-world IoT datasets	Improved IoT data sharing security and clustering performance	-

3 Materials and methods

3.1 Research design

This study aims to address the challenge of balancing data sharing security and clustering accuracy in heterogeneous IoT environments. To this end, the following research questions are addressed:

RQ1: In federated scenarios, can the integration of SL and fuzzy multi-objective decision-making methods significantly improve clustering accuracy and sample identification quality on heterogeneous IoT data?

RQ2: Does SecureBoost homomorphic encryption enhance security and privacy protection without compromising clustering performance?

To answer these questions, the USPS and synthetic datasets were used as benchmarks to verify the algorithm's generalization and adaptability to complex environments, respectively. Evaluation metrics included classification accuracy (ACC), normalized mutual information (NMI), F1 score, ARI, transaction latency, and encryption time. The system architecture consists of a SL module, a fuzzy multi-objective decision-making module, and a SecureBoost encryption aggregation module. Blockchain and IPFS technologies are used to implement a secure sharing environment. The experimental design and evaluation indicators are centered around RQ1 and RQ2 to ensure the consistency and verifiability of the research objectives, methods, and results.

3.2 Optimization of clustering federated learning algorithm based on SL improvement

With the continuous advancement of IoT applications, the number of smart terminals in various fields such as education and economy has also increased significantly [15]. However, during the process of IoT machine learning, issues such as poor data clustering and unstable feature extraction often arise [16]. To address these challenges, this study proposes a Clustering Federated Learning algorithm based on SL to optimize data dimensionality reduction and clustering. SL combines limited labelled data with abundant unlabelled data to improve model generalization while maintaining privacy. Even when data identification is unclear, SL can effectively label unlabelled information [17]. SL clustering suits federated learning by improving clustering quality with limited labelled information while preserving privacy [18]. Data heterogeneity across clients can cause semantic inconsistencies in unsupervised approaches, but supervised signals guide local models toward consistency without exposing raw data, enhancing global model aggregation [19]. The improved average clustering distance d_k using SL is shown in Equation (1).

$$d_k = \frac{1}{n_k - 1} \sum_{\left(x_i^{(k)} \in C_k, x_i^k \neq u_k\right)} d\left(x_i^k, u_k\right) \quad (1)$$

In Equation (1), C represents the total number of clusters, u_k denotes the cluster centers, n_k and C_k represent the number of samples and the sample set in k clusters, respectively, and $x_i^{(k)}$ represents the i-th sample excluding the cluster center. The descending order of the sample membership degrees is expressed in Equation (2).

$$\Delta h^{(s)} = h_o^{(s)} - h_t^{(s)} \quad (2)$$

In Equation (2), $\Delta h^{(s)}$ represents the sample

membership. s represents individual samples in the outer region set, while $h_o^{(s)}$ and $h_t^{(s)}$ denote the maximum membership degree and the second-largest membership degree, respectively. The sample allocation based on the improved SL is shown in Figure 1.

In Figure 1, during the self-training rounds, the confidence of samples in the inner region is higher, while the confidence of samples in the outer region is weaker. To ensure that the training samples meet the target labeling, the confidence of samples within the region is calculated separately. The sample with the highest confidence is selected for prediction, and a classifier is used for labeling. The SL method also handles samples with small membership differences to obtain high-quality samples. The fuzzy multi-objective decision-making method performs clustering on IoT data based on the SL algorithm. Fuzzy multi-objective decision-making is a method that combines fuzzy mathematics and multi-objective optimisation to make trade-off choices in situations involving multiple evaluation indicators or uncertainties. The collaborative filtering function for IoT event correlation features obtained using the fuzzy multi-objective decision-making method is shown in Equation (3).

$$\sigma(A,B)=S_i\times\frac{(H_i\times Q_i)\frac{1}{|p_i|}}{G(A,B)|u|} \quad (3)$$

In Equation (3), $\sigma(A,B)$ represents the mapping function, i denotes the number of indicators. $|p_i|$ represents the mapping imaginary number, which indicates that the input image will be encoded through a mapping function in the complex domain to enhance style expression capabilities. S_i represents the total samples in the external region set. And H_i indicates the similarity index. A and B represent the amplitude adjustment coefficients for the real part and imaginary part, respectively, during the transformation process. The IoT data clustering process improved by the fuzzy multi-objective decision-making method is shown in Figure 2.

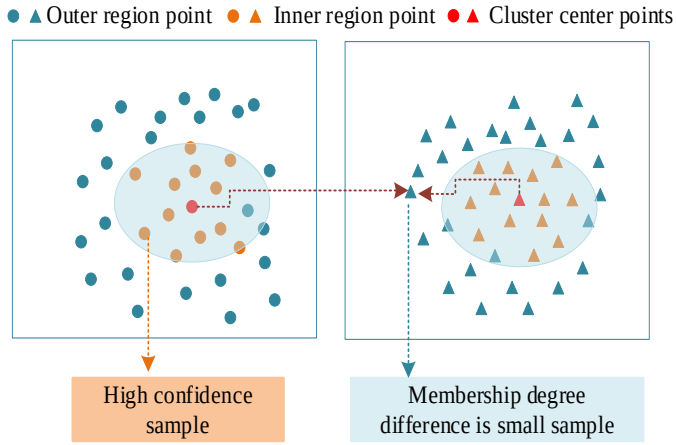


Figure 1: Sample distribution diagram based on SL improvement

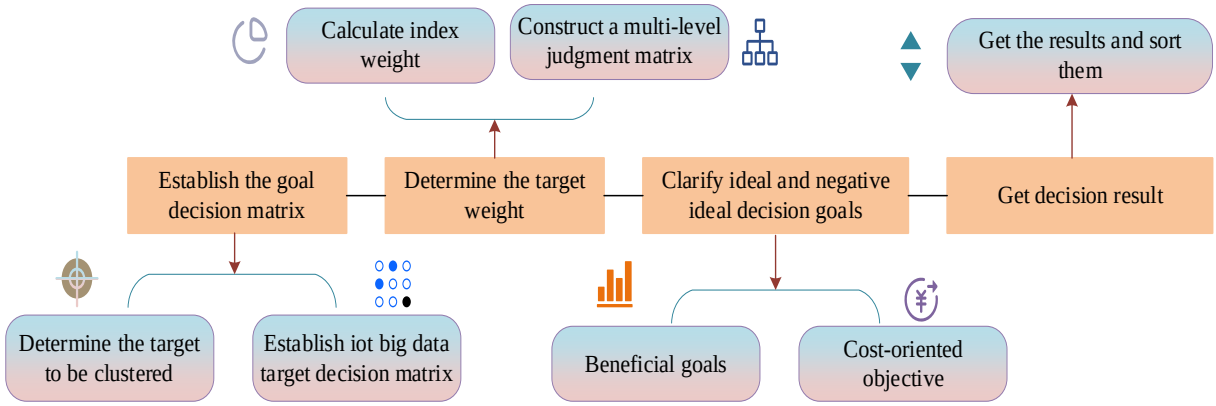


Figure 2: Improved IoT data clustering flow chart

As shown in Figure 2, the method first establishes an objective decision matrix and determines clustering objectives. After constructing the objective decision matrix, the weights of the objectives are determined, and the IoT clustering evaluation indicator weights are calculated. The scale values are determined hierarchically, and a multi-level judgment matrix is constructed. The benefit-type and cost-type objectives are categorized

when determining ideal and negative ideal objectives. Finally, the decision results are used to establish a dataset, and the membership degrees are arranged in order. The entropy weight of IoT clustering evaluation indicators is shown in Equation (4).

$$Z_i = \frac{\gamma_i}{\frac{1}{I}\sum_{i=1}^{I}\gamma_i} \quad (4)$$

In Equation (4), i represents the indicator, Z_i denotes the entropy weight, and γ_i represents the IoT data correlation entropy weight value. The modularity calculation for joint clustering is shown in Equation (5).

$$Q(w,C) = Q\left(w, U \cdot V^T\right) = \frac{1}{W}\sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{l}^{k}\left(a_{ij} - \frac{a_{i\cdot}a_{\cdot j}}{W}\right)u_{il}v_{jl} \quad (5)$$

In Equation (5), W represents the sum of all weights in the data matrix w, $a_{i\cdot}$ and $a_{\cdot j}$ denote the sum of weights in the i-th row and j-th column, respectively. U and V represent the row and column indicator matrices, and l represents the cluster. The cross-entropy loss calculation for IoT image clustering is shown in Equation (6).

$$L_{CE} = \sum_{j=1}^{m}\sum_{i=1}^{n} -y_{ji}\log\left(\hat{y}_{ji}\right) - \left(1-y_{ji}\right)\log\left(1-\hat{y}_{ji}\right) \quad (6)$$

In Equation (6), L_{CE} represents the IoT cross-entropy loss, m and n denote the number of image categories and the number after classification, respectively, and y_{ji} and $\hat{y}_{ji}$ represent the true labels and final output of the network. This paper adopts a pseudo-label mechanism to adapt cross-entropy loss to clustering tasks. Pseudo-labels are generated from model predictions and dynamically updated to provide stable optimization targets and enhance feature clustering properties. The specific workflow of the improved Clustering Federated Learning algorithm based on SL is shown in Figure 3.

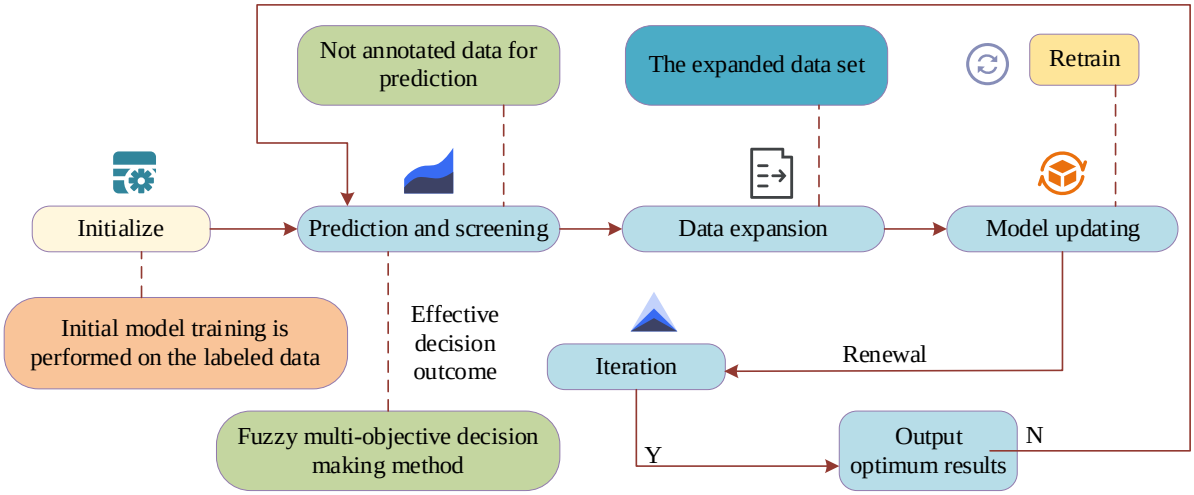


Figure 3: Flowchart of clustering joint learning algorithm based on SL improvement

As shown in Figure 3, the improved Clustering Federated Learning algorithm based on SL first selects partially labeled IoT data for initial model training. The trained model is then used to predict and filter unlabeled data. During this process, the fuzzy multi-objective decision-making method is employed to extract key data features from multiple objectives, and a confidence threshold is set to label the objectives, selecting the optimal decision results [20]. The filtered and predicted data are then used to expand the dataset, forming a new dataset. The new dataset retrains the model, and the resulting model is iterated. If the iteration training count is reached, the optimal result is output; otherwise, the data filtering and prediction process is repeated. This paper integrates fuzzy multi-objective decision-making with federated learning by introducing a fuzzy clustering process in each participant. Each participant first constructs a multi-objective decision matrix based on local data and calculates fuzzy weights and membership degrees to extract more discriminative feature representations. Subsequently, high-membership-degree samples are used for semi-supervised pseudo-label generation to improve the quality of the supervisory signal in the local model. During the federated aggregation phase, each client uploads the model parameters optimised based on fuzzy clustering, and the server performs weighted averaging for model aggregation.

To enhance pseudo-label reliability and aggregation stability, each node integrates fuzzy multi-objective decision-making with SecureBoost encryption. In each training round, participants construct a fuzzy decision matrix, calculate membership and weights, and select high-confidence samples. In this paper, the confidence threshold is set to 0.82 through grid search and 5-fold cross-validation. Low-confidence samples (membership > 0.5) are processed using an adaptive soft labeling and sliding window relabeling strategy, and their pseudo-labels are updated in up to three iterations. If the threshold

is still not met, the sample is excluded from the global model update in the current round. Samples with membership ≤ 0.5 are directly eliminated. To account for potential distribution heterogeneity among federated nodes, the study further employs the SecureBoost mechanism to homomorphically encrypt the gradient and Hessian information after pseudo-label generation, ensuring that split point calculation and decision tree construction are completed without exposing the original features and labels. The parameters uploaded by each participant only contain encrypted gradient statistics. The server performs global model aggregation through weighted averaging, thereby reducing the problem of pseudo-label shift caused by differences in node distribution. In addition, by introducing fuzzy weights to dynamically adjust node contributions, high-quality nodes are given greater weight in the aggregation, mitigating the impact of heterogeneity on model performance.

3.3 Enhancement of IoT data sharing technology method integrating SL and SecureBoost

IoT stores massive amounts of data, bringing significant commercial value and innovative industrial opportunities to various sectors [21]. However, the rapid development of IoT has also introduced issues such as unreliable shared data and malicious interference with data sources [22]. Although the improved SL-FL algorithm aggregates different IoT data types, it has limitations in secure data sharing. To address privacy and leakage issues, this study proposes a SecureBoost-based data sharing technology built on the SL-FL algorithm. SecureBoost is implemented on FATE framework (version 1.10.0) with lightweight modifications for IoT scenarios. SecureBoost uses Paillier homomorphic encryption to encrypt sensitive gradient and Hessian information, enabling secure interaction between multiple participants. During training, the active participant with the label locally encrypts the gradient and Hessian before initiating decision tree construction. Other participants only need to upload encrypted feature statistics to complete feature splitting and model training. Throughout the entire process, the server only processes ciphertext aggregation and does not access any plaintext data, effectively protecting the data privacy of all participants. SecureBoost is shown in Equation (7).

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), \forall x \in R^d, i = 1,2,3,...,n \quad (7)$$

In Equation (7), n and d represent samples and features, respectively, f_k denotes the decision tree, and $x_i \in R^d$ and $y_i \in R$ are parameters. The objective function is shown in Equation (8).

$$obj = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{i=1}^{n} \Omega(f_k) \quad (8)$$

In Equation (8), y_i and $\hat{y}_i$ represent the true label values and model-predicted values, respectively, $\Omega(f_k)$ denotes the penalty term, and $l(y_i, \hat{y}_i)$ represents the loss function. The multi-party interaction framework for federated learning is shown in Figure 4.

As shown in Figure 4, the multi-party interaction framework for federated learning based on SecureBoost is divided into active and passive parties, enabling privacy-preserving data sharing between them. Participants exchange information through encrypted communication and security protocols to align entity samples while preserving privacy. Then, Local Model A and Local Model B are used to construct a global boosting tree, and the predicted output is obtained by calculating the weights of the decision trees. Finally, the decision tree model is updated. The threshold for node splitting in the decision tree is shown in Equation (9).

$$L_{split} = \frac{1}{2}\left[\frac{\left(\sum i \in I_L g_i\right)^2}{\sum i \in I_L h_i + \lambda} + \frac{\left(\sum i \in I_R g_i\right)^2}{\sum i \in I_R h_i + \lambda} - \frac{\left(\sum i \in I g_i\right)^2}{\sum i \in I h_i + \lambda}\right] \quad (9)$$

In Equation (9), I_L represents the sample space of the left child node after splitting, I_R denotes the sample space of the right child node after splitting, λ represents the hyperparameter, g_i is the first-order gradient of the loss function, and h_i denotes the second-order gradient. The study also utilizes blockchain technology to ensure transaction fairness and security. Blockchain technology has features such as tamper resistance, ensuring that transaction data is transparent and secure. The specific structure of the data layer for transactions using blockchain technology is shown in Figure 5.

As shown in Figure 5, the blockchain transaction data layer is formed by connecting multiple blocks. The block header contains metadata such as version number, timestamp, difficulty, and nonce for verifying data integrity and computing consensus mechanisms. Each block is linked to the hash value of its parent block, forming a chain-like data structure. This structure is traceable and tamper-resistant. The transaction data in the block body is hashed, and the Merkle root hash is stored in the block header. If data tampering is suspected, it can be quickly verified using the Merkle tree. The bilinear mapping calculation for IoT user requests for data sharing is shown in Equation (10).

$$G_1 \times e \to G_2 \quad (10)$$

In Equation (10), G_1 and G_2 represent the user's digital signature and mapping result, respectively, and e denotes the bilinear mapping function. The random generation of unique user identity information is shown in Equation (11).

$$S = G_2 \times sP_{i,j} \quad (11)$$

In Equation (11), i and j represent the IoT environment node and random number, respectively, while P and s denote the forged signature and IoT identity recognition consensus. The terminal request when

a user needs to perform IoT data sharing transmission is shown in Equation (12).

$$Y = P\left(H\left(I\right) + SK\right) \quad (12)$$

In Equation (12), Y and H represent the data request and command conditions, respectively, K denotes the data transmission link during the request, and I represents the final verification request for execution. The resource request and access process for the improved IoT data sharing technology based on the SL-FL and SecureBoost is shown in Figure 6.

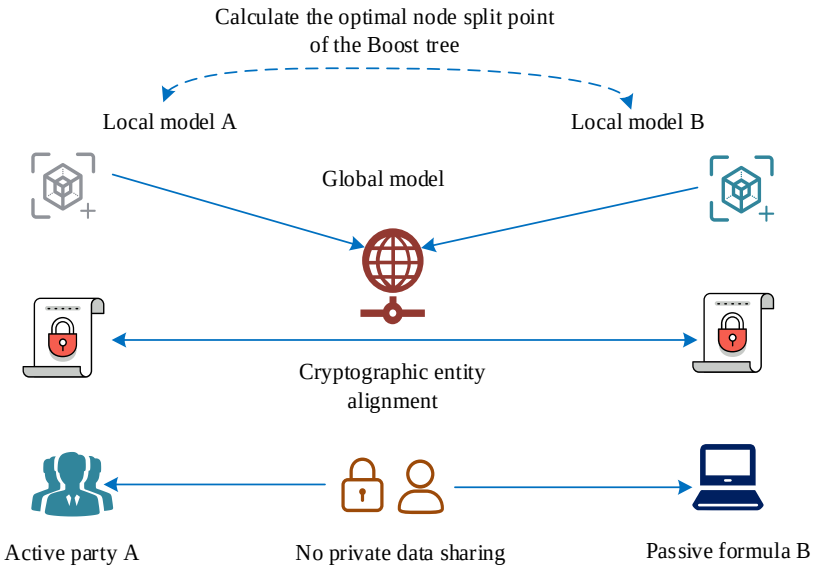


Figure 4: Federated learning multi-party interactive framework diagram

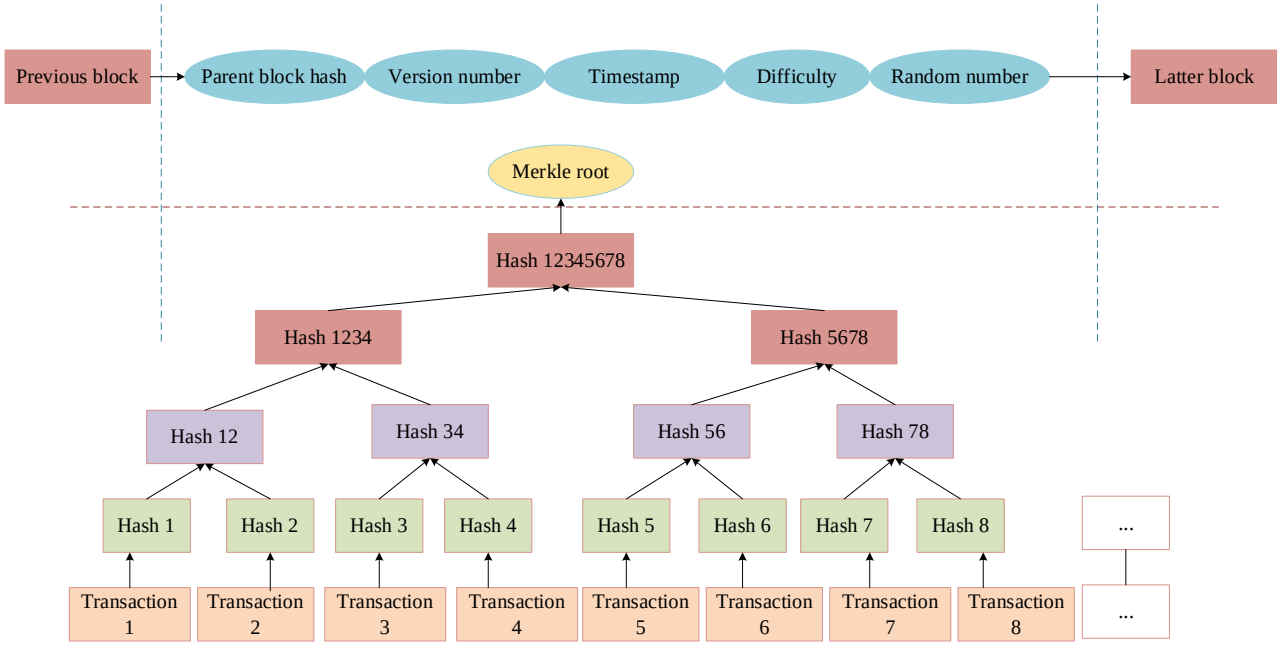


Figure 5: Specific structure diagram of the data layer

As shown in Figure 6, the access request first obtains user information from the IoT environment. A distributed storage protocol is created and the information is transmitted to the user via IPFS. Then, the user stores address information through the blockchain network and makes a request for access. The blockchain network requires approval from the IoT resource library and administrators, after which the user's information is verified. The request command obtains the resource's public address through the blockchain network. During the data sharing request, the SL-FL and SecureBoost are combined for data aggregation and privacy protection. Finally, the request is recorded by the IPFS system, and the requested data is transmitted to the user. A simplified example demonstrates the synergy between SL, fuzzy multi-objective decision-making, and SecureBoost.

Assume a system with three samples (A, B, and C) and two participating nodes. First, SL calculates membership based on the initial labels, filtering A and B as high-confidence samples (membership ≥ 0.82) and C as a low-confidence sample. Subsequently, fuzzy multi-objective decision-making includes A and B in the pseudo-label set to enhance supervision. Next, the active node uses Paillier homomorphic encryption on the gradients and Hessians of A and B and initiates decision tree construction. The passive node only uploads the corresponding encrypted feature statistics. The server then aggregates the ciphertext and performs a split operation.

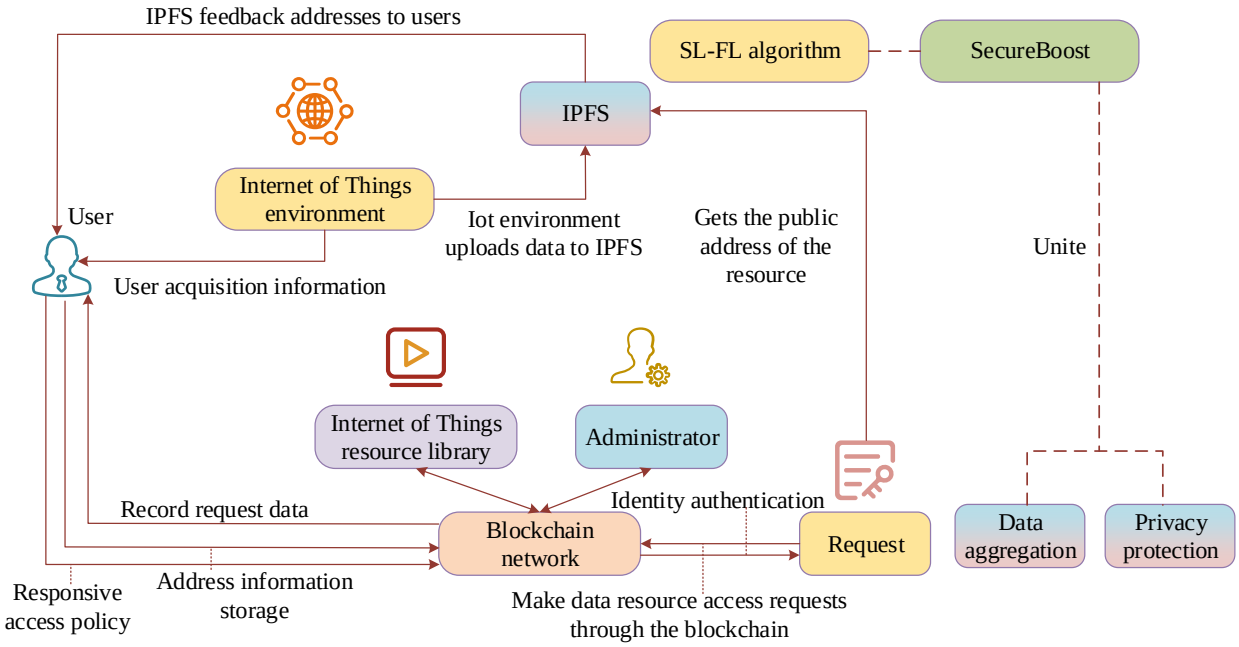


Figure 6: IoT data sharing technical resource request access flow chart

4 Experimental and results

4.1 Experimental setup

Experiments were conducted in an environment built on the Spring Boot framework, using Windows 11 as the operating system, 16 GB of RAM, an Intel(R) i5-6402P CPU, and Java as the programming language. IoT experiments were conducted using Red Hat Gigabit network equipment. The datasets used include the USPS dataset and synthetic datasets, as well as FATE's internal dataset and publicly available IoT datasets from Kaggle. The USPS dataset comprises 9,298 16×16 grayscale images (7,291 training, 2,007 test), frequently used for evaluating clustering performance due to its clear structure and complex feature distribution. The synthetic dataset comprises 4,000 two-dimensional sample points categorized into four classes, each containing 1,000 samples. Class centers are at (0,0), (0,40), (40,0), and (40,40), with Gaussian noise (mean=0, variance=4) added. During data preprocessing, missing values were imputed and outliers were removed. Numerical features were normalized using Z-scores, and categorical features were converted to numerical form using one-hot encoding. Principal component analysis was used to reduce the dimensionality of some high-dimensional features to improve clustering efficiency. Model training utilizes the SecureBoost encrypted federated clustering framework. Each participant performs SL clustering and fuzzy multi-objective decision-making locally, while the server completes ciphertext aggregation. Core hyperparameter settings are shown in Table 2.

Table 2: Hyperparameter settings

Parameter name	Value
Learning rate	0.01
Batch size	64
Local epochs	5
Global rounds	20
Confidence threshold	0.82
Maximum number of re-labeling rounds	3
SecureBoost max tree depth)	5
Encryption scheme	Paillier homomorphic encryption
Loss function)	Logistic Loss

4.2 Effectiveness verification of SL-FL

To verify the superior performance of the SL-FL algorithm, the study compared it with the Multi-View Deep Subspace Clustering Networks (MvDSCN) algorithm, Mid-level Deep Pattern Mining (MDPM) algorithm, and Enhanced Weighted Fuzzy C-Means (EWFCM) algorithm. The results are shown in Figure 7. As shown in Figure 7(a), there were four different types of clusters before classification, with cluster centers at (0,0), (0,40), (40,0), and (40,40). From Figure 7(b), the SL-FL algorithm positioned Cluster 1 center at (0,39) and Cluster 2 at (40,38), showing minimal deviation from true centers. From Figure 7(c), after clustering by the MvDSCN

algorithm, some data points from Cluster 3 were incorrectly identified as belonging to Cluster 1, with the misclassified region centered around (10,25). From Figure 7(d), the clustering performance of the MDPM algorithm was inferior to that of the SL-FL algorithm, with the center of Cluster 4 at (37,5) and a small number of data points incorrectly assigned to Cluster 2. From Figure 7(e), the EWFCM algorithm performed worse than the other three algorithms, with some data points from Cluster 3 and Cluster 4 incorrectly assigned to Cluster 1 and Cluster 2, and the misclassified region for Cluster 4 centered around (30,35). In summary, the SL-FL algorithm demonstrated the best clustering performance. To further demonstrate the clustering performance of the SL-FL, the study compared it with the MvDSCN algorithm, MDPM algorithm, and EWFCM algorithm on the USPS dataset and the synthetic dataset using four metrics: Classification Accuracy (ACC), Normalized Mutual Information (NMI), F-score, and Adjusted Rand Index (ARI). The test results are shown in Table 3.

From Table 3, on the USPS dataset, the SL-FL algorithm achieved an ACC of 92.4%, NMI of 0.846, F-score of 0.876, and ARI of 0.965. The MvDSCN algorithm achieved an NMI of 0.836 and ACC of 89.3% on the USPS dataset, both lower than SL-FL. When tested on the synthetic dataset, the EWFCM algorithm achieved an ACC of 79.6% and an NMI of 0.810, which was 15.0% lower than the ACC of the SL-FL algorithm. In summary, the SL-FL algorithm demonstrated excellent clustering and classification performance across different datasets and effectively measured overall data performance. Additionally, the study conducted Macro-F1 tests on the SL-FL algorithm, MvDSCN, MDPM, and EWFCM algorithms. The results are shown in Figure 8.

As shown in Figure 8(a), when testing Classification Task 1, the SL-FL algorithm achieved a Macro-F1 score of 81.23%, while for Classification Task 4, the Macro-F1 score was 97.68%. Among all tasks, Classification Task 4 achieved the highest Macro-F1 score. From Figure 8(b), when testing Classification Task 3, the SL-FL algorithm achieved a Macro-F1 score of 89.68%, which was 17.01% higher than the 72.67% achieved by the MvDSCN algorithm. For Classification Task 1, the MDPM algorithm achieved a Macro-F1 score of 68.34%, while for Classification Task 4, the score was 57.67%. In summary, the SL-FL performed consistently well across different classification tasks.

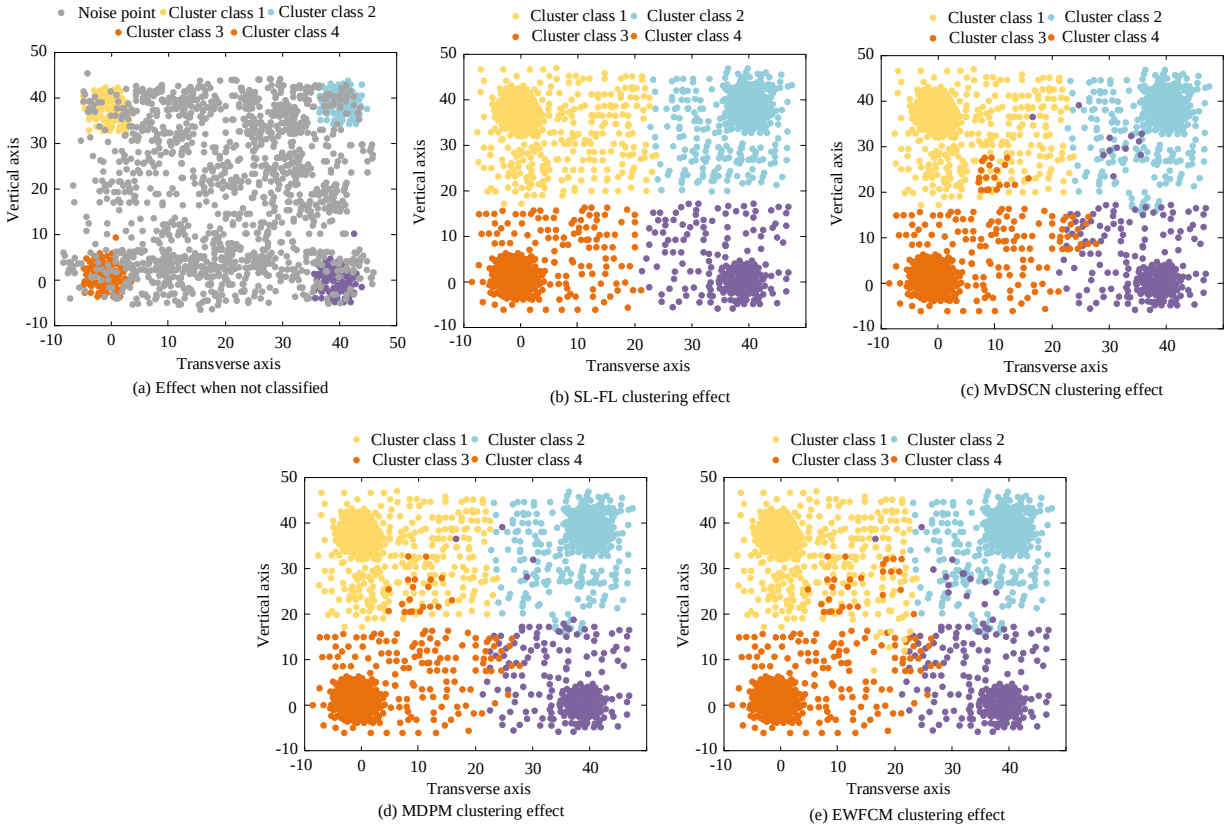
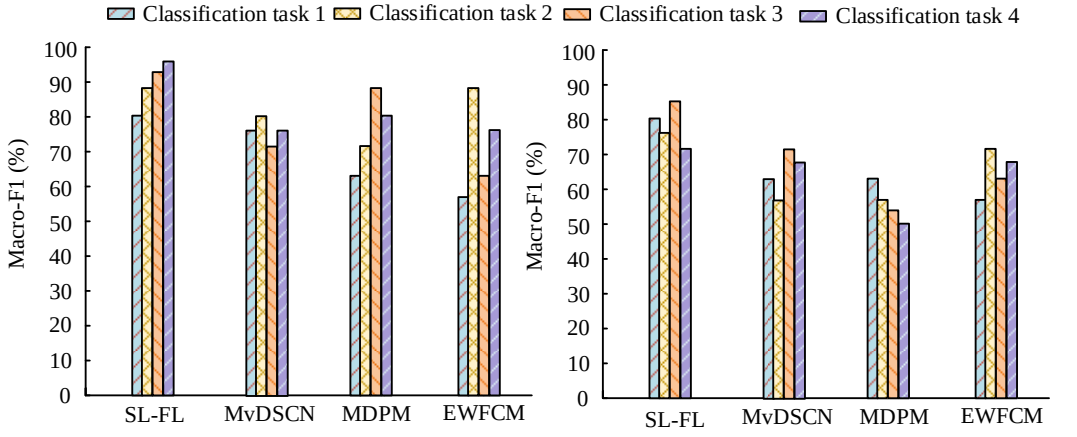


Figure 7: Clustering effect test results

Table 3: Test results of different algorithms in USPS and synthetic data sets respectively

Data set	Algorithm	ACC (%)	NMI	F-score	ARI
USPS	SL-FL	92.4	0.846	0.876	0.965
	MvDSCN	89.3	0.836	0.821	0.867
	MDPM	79.6	0.803	0.846	0.816
	EWFCM	78.4	0.796	0.863	0.896

Synthetic data sets	SL-FL	94.6	0.896	0.914	0.953
	MvDSCN	90.3	0.897	0.768	0.897
	MDPM	84.6	0.814	0.869	0.901
	EWFCM	79.6	0.810	0.836	0.865



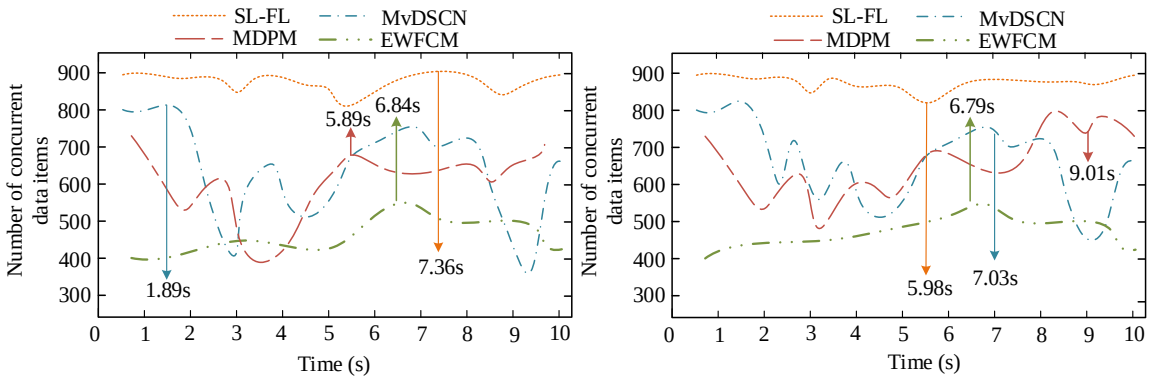
(a) Macro-F1 indicator test results in USPS data set

(b) Macro-F1 index test results of a synthetic dataset

Figure 8: Multi-classification F1 value test results

4.3 Evaluation and analysis of improved SL IoT data sharing fusion technology

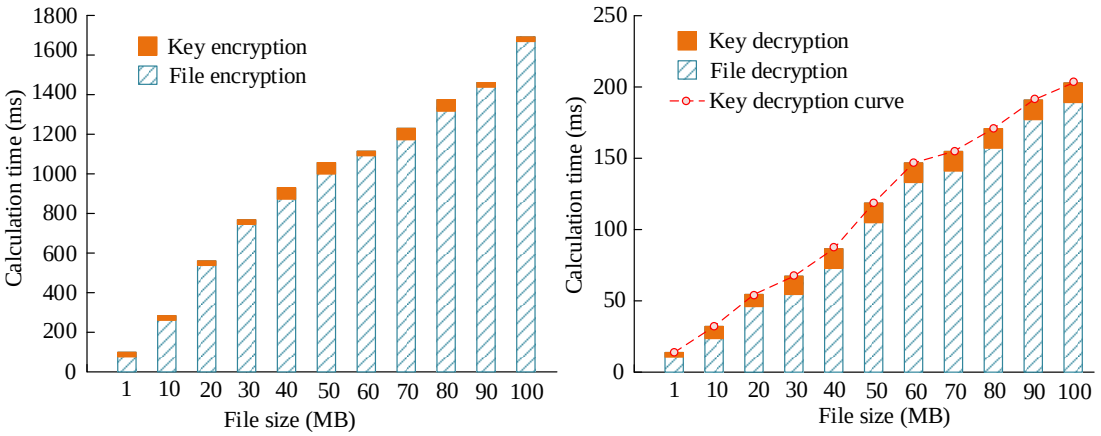
After verifying the performance of the SL-FL, this study further analyzed the performance of the IoT data sharing and fusion technology proposed based on this algorithm and compared it with IoT data technologies based on the MvDSCN, MDPM, and EWFCM algorithms. The comparison results are shown in Figure 9.



(a) Efficiency test results are shared in FATE internal datasets

(b) Sharing efficiency test results in the Kaggle IoT dataset

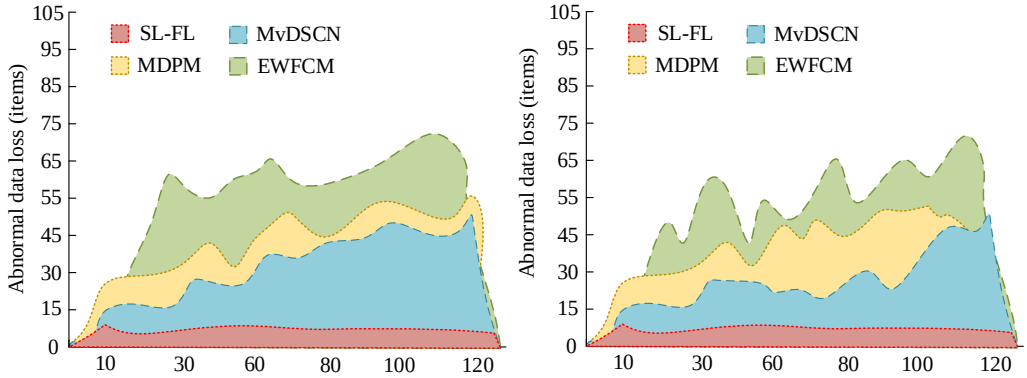
Figure 9: IoT data sharing technology sharing efficiency test results



(a) Symmetric key time comparison for different file sizes

(b) Symmetric decryption times for different file sizes

Figure 10: File symmetric encryption and decryption test results



(a) Data sharing and transmission of test results within the FATE internal dataset

(b) Data sharing transfer test results within the Kaggle IoT dataset

Figure 11 Resource data sharing test results

As shown in Figure 9(a), on the FATE internal dataset, the SL-FL processed 910 concurrent requests in 7.36 s, the highest among all methods. The MvDSCN IoT data sharing technology processed 400 to 420 front-end concurrent requests in 3 to 4 s. When processing 520 concurrent requests, the EWFCM took 6.84 s. From Figure 9(b), the SL-FL was capable of handling multiple data processes, processing 863 requests in 5.98 s. The MDPM showed unstable processing times and processed 736 requests in 9.01 s. In summary, the SL-FL demonstrated superior concurrent processing capabilities. Additionally, the study conducted symmetric encryption and decryption tests on the SL-FL. The results are shown in Figure 10.

As shown in Figure 10(a), for files smaller than 10 MB, the key encryption time was within 250 ms. For a 40 MB file, the key encryption time was 892 ms, which was 94 ms longer than the time for a 30 MB file. As the file size increased, the encryption time also increased, reaching 1678 ms for a 100 MB file. From Figure 10(b), for a 1MB file, the decryption time was 19 ms. For 60 MB and 70 MB files, the key decryption times were similar, at 149 ms and 152 ms, respectively. Overall, decryption time increased with file size. To further demonstrate the resource sharing and transmission performance of the SL IoT data sharing fusion technology, the study compared it with the MvDSCN, MDPM, and EWFCM IoT data sharing technologies. The test results are shown in Figure 11.

As shown in Figure 11(a), the SL-FL lost fewer than 15 data packets and maintained stable performance throughout the resource sharing process. The MvDSCN lost 49 data packets after 100 resource sharing transmissions. From Figure 11(b), the EWFCM lost a significant number of data packets during the resource sharing process, with 62 packets lost after 30 transmissions. The MDPM lost 27 to 30 data packets during 10 to 30 transmissions. In summary, the SL-FL maintained lower packet loss during resource sharing.

4.4 Performance comparison and security analysis

To further validate the advantages of the proposed method in terms of security and performance, the study compares it with blockchain-based sharing, differential privacy federated learning (DP-Fed), EWFCM, secure multi-party computation federated learning (SMPC-FL), and homomorphic encryption federated learning (HE-FL). The experimental results cover multiple metrics, including data sharing efficiency, packet loss rate, encryption time,

and ACC. Packet loss monitoring is accomplished using the Wireshark capture tool and server-side log statistics, recording the total number of packets sent and received within a fixed time window. Data sharing efficiency refers to the number of requests successfully processed per unit time. Concurrent requests are simulated using the JMeter stress testing tool, and responses are recorded using Spring Boot Metrics + Prometheus to calculate average throughput performance. Paired t-tests were used for significance analysis. Each experiment was repeated five times independently, with the mean and standard deviation calculated. A significance level of $p < 0.05$ was set as the threshold for determining significant differences. In addition, in order to evaluate the security protection capabilities of different methods, three typical scenarios were designed: replay attack, eavesdropping attack, and malicious node attack, and the attack success rate was used as the evaluation indicator. The specific results are shown in Table 4.

As shown in Table 4, SL-FL's data sharing efficiency improves by 9.25%, 9.71%, 23.99%, 29.96%, and 67.34% compared to the blockchain sharing mechanism, DP-Fed, SMPC-FL, HE-FL, and EWFCM, respectively. These differences are statistically significant ($p<0.05$), demonstrating superior performance in high-concurrency environments. In terms of encryption latency, SL-FL's encryption time is 209.17 ms, significantly lower than the other five methods, demonstrating a good balance between security and efficiency. In terms of ACC, SL-FL achieves an accuracy of 94.43%, slightly higher than the blockchain sharing mechanism by 0.14%, slightly lower than DP-Fed by 0.19%, and significantly higher than SMPC-FL, HE-FL, and EWFCM ($p<0.05$). This demonstrates that SL-FL maintains model accuracy while preserving privacy, though slightly lower than DP-Fed due to noise injection. In terms of packet loss rate, SSL-FL demonstrated greater transmission stability and network adaptability. In security attack experiments, SL-FL achieved a success rate of only 2.13%, significantly lower than the other five methods ($p<0.05$). This demonstrates that SL-FL, by combining lightweight homomorphic encryption with blockchain traceability, significantly enhances its ability to resist eavesdropping, replay, and malicious node attacks while maintaining privacy.

Table 4: Comparison of the performance of different IoT security methods

Method	Data sharing efficiency (req/s)	Encryption time (ms)	ACC (%)	Packet loss rate (%)	Attack success rate (%)
SL-FL	146.65±7.34a	209.17±8.66a	94.43±0.16a	1.14±0.12a	2.13±0.47a
Blockchain sharing	134.25±6.80b	215.21±5.52b	94.29±1.15a	1.29±0.23b	6.42±1.18b
DP-Fed	133.64±5.10b	226.24±7.99c	94.61±1.25a	1.37±0.25b	5.83±0.91b
SMPC-FL	118.29±4.87c	282.43±11.22c	93.82±1.31b	1.26±0.18a	2.76±0.82a
HE-FL	112.86±4.64c	300.27±12.54c	93.51±1.48b	1.21±0.17a	2.98±0.73a
EWFCM	87.67±5.20d	-	79.63±1.25c	1.45±0.31b	21.74±2.37c

Note: The values with different lowercase letters (a, b, c) in the same column indicate significant differences ($p < 0.05$), and the same letters indicate no significant differences.

4.5 Scalability verification

To further assess the scalability and modular contributions of the proposed SL-FL framework in practical IoT scenarios, scalability experiments were conducted using a real-world IoT sensor dataset. The dataset originates from an industrial IoT monitoring scenario, where data is continuously collected by an edge gateway and uploaded to a cloud server. The dataset contains over 5,000 edge nodes from 38 types of industrial equipment, including temperature and humidity sensors, pressure sensors, flow meters, and power acquisition modules. The sampling frequency is 1 Hz, and the total data volume is approximately 2.3 TB, spanning 45 days. Sensors are distributed across multiple areas of the factory, and the network topology exhibits a multi-layered star and mesh hybrid structure. Link status fluctuates significantly, resulting in high latency, jitter, and packet loss. Table 5 shows the ablation validation of the complete SL-FL framework on the real-world dataset.

Table 5 shows that adding the SL module increases throughput to 132.21 req/s, improves ACC by 4.22%, and reduces packet loss by 0.13%, demonstrating that the semi-supervised mechanism enhances feature extraction and clustering. Adding fuzzy decision-making further improves throughput, reaching 94.21% and reducing packet loss to 1.18%, demonstrating its adaptability to uncertain data. With the introduction of SecureBoost, throughput reaches a maximum of 146.65 req/s, maintaining ACC at 94.43%, enhancing privacy protection without significantly impacting performance. Finally, adding the blockchain module reduces throughput slightly to 143.12 req/s, further reducing packet loss and maintaining stable ACC. While blockchain introduces a slight overhead, it improves the system's anti-tampering and security capabilities, achieving a good balance between performance, privacy, and security.

To further evaluate the security and scalability of the proposed framework in real-world deployment scenarios, the blockchain module was tested separately. The blockchain node size was adjusted (20, 50, 100, and 200), and transaction latency, block generation rate, and consensus time were recorded. Two typical attack scenarios were configured: (1) a double-spending attack, simulating malicious nodes repeatedly broadcasting transactions to disrupt ledger consistency; and (2) a Sybil attack, in which fake nodes are used to interfere with the

consensus process. The attack success rate is defined as the ratio of successful attacks to the total number of attempts. The results are shown in Table 6.

Table 5: Ablation verification

Configuration	Throughput (req/s)	Packet loss rate (%)	ACC (%)
(i) FL baseline	123.82	1.42	86.52
(ii)+SL	132.21	1.29	90.74
(iii) + SL + Fuzzy Decision	140.15	1.18	94.21
(iv) + SL + Fuzzy Decision + SecureBoost	146.65	1.14	94.43
(v) + Blockchain (Complete SL-FL framework)	143.12	1.12	94.43

Table 6: Blockchain scalability and attack model verification under different network scales

Number of nodes	Transaction delay (ms)	Block generation rate (block/s)	Consensus time (ms)	Double-spending attack success rate (%)	Sybil attack success rate (%)
20	72.45	1.12	65.33	0.00	0.10
50	85.12	1.05	78.42	0.00	0.18
100	103.67	0.98	94.56	0.10	0.26
200	128.94	0.91	116.73	0.22	0.41

Table 6 shows that when the number of nodes increases from 20 to 200, transaction latency increases to 128.94 ms, while the block generation rate decreases to 0.91 blocks/s, which is consistent with the latency characteristics. Despite a slight increase in overhead, the success rate of double-spending and Sybil attacks remains below 0.5%, demonstrating stable security. This demonstrates that the blockchain module possesses strong scalability and security protection capabilities when expanded to large-scale networks. Its low communication overhead yields greater attack resistance and trustworthiness, further enhancing the feasibility of the entire SL-FL framework for deployment in real-world IoT scenarios.

5 Discussion

The experimental results demonstrate that SL-FL achieves superior throughput, packet loss rate, and security protection while maintaining high accuracy. In the basic performance evaluation, the throughput of SL-FL improved by 67.34%, 29.96%, and 23.99% compared with EWFCM, HE-FL, and SMPC-FL, respectively. This indicates that the lightweight encryption and fuzzy pseudo-labeling mechanisms effectively reduce communication and computation overhead in high-concurrency scenarios. These findings align with references [12] and [14], which report similar throughput improvements through lightweight security mechanisms. Compared with DP-Fed, the accuracy of SL-FL is slightly lower, but its throughput and packet loss rate are significantly better. This is because DP-Fed's stronger differential privacy noise increases communication and computation overhead, reducing performance. This observation aligns with the analysis of the trade-off between privacy strength and accuracy in reference [18]. In contrast, SL-FL maintains high accuracy with greater efficiency and stability, making it more suitable for dynamic IoT scenarios. The lower ACC of EWFCM in heterogeneous IoT environments is mainly caused by noise interference and boundary ambiguity, which is consistent with the instability of fuzzy clustering algorithms in complex environments reported in reference [21]. Although HE-FL and SMPC-FL provide strong privacy protection, their high encryption and communication costs reduce throughput, consistent with reference [22].

In scalability evaluation, increasing nodes from 20 to 200 caused only slight throughput decrease and moderate latency increase, while attack success rate remained below 0.5% with stable block generation. This demonstrates the blockchain module's scalability and security in large-scale IoT networks, consistent with references [28] and [29]. In terms of communication cost, the lightweight SecureBoost encryption and fuzzy mechanism effectively reduce average encryption latency, cutting more than 30% of the overhead compared with HE-FL and SMPC-FL. However, increasing nodes intensifies synchronization pressure between edge and central nodes, potentially causing bandwidth consumption and resource imbalance. This finding is consistent with the conclusions in references [14] and [18], which highlight that the coupling of encrypted computation and high-frequency communication leads to reduced efficiency. Introducing compression aggregation or dynamic scheduling mechanisms can further optimize communication cost. Deployment in dynamic IoT environments remains challenging. Frequent node fluctuations and unstable link quality require the system to support rapid fault tolerance and topology adaptation. Although SL-FL enhances robustness, performance degradation may still occur under extreme conditions. References [31] and [32] indicate that a single security or learning mechanism is insufficient to address complex dynamic environments, and combining adaptive control with hierarchical scheduling could further improve system adaptability and

stability.

6 Conclusion

To address privacy protection and data aggregation challenges in IoT data sharing, the study proposed a hybrid technology based on SL-FL to improve resource sharing efficiency. The technology utilizes fuzzy multi-objective decision-making for clustering and SecureBoost for homomorphic encryption, protecting the data transmission process. Experimental results showed that SL-FL achieved a Macro-F1 score of 97.68% for Classification Task 4, outperforming MvDSCN (78.92%), MDPM (82.36%), and EWFCM (79.14%). Additionally, empirical analysis revealed that the SL IoT hybrid sharing technology lost fewer than 15 data packets during 120 sharing transmissions, while the MvDSCN, MDPM, and EWFCM IoT data sharing technologies lost more than 25 data packets and exhibited unstable sharing performance. SL-FL significantly outperforms EWFCM, HE-FL, SMPC-FL, and DP-Fed in terms of throughput, packet loss rate, and attack protection capability, and is close to DP-Fed in terms of accuracy, only slightly lower by 0.19%. In summary, the IoT data sharing hybrid technology based on the SL-FL improved data sharing security and effectively classified different types of data.

While SL-FL demonstrates outstanding throughput, packet loss rate, and security performance, it still has limitations. First, its accuracy is slightly lower than DP-Fed, reflecting a trade-off between privacy protection and accuracy. Second, the experiments were primarily conducted in a controlled environment, and its adaptability to large-scale heterogeneous and dynamic networks has not yet been fully verified. Furthermore, the framework requires improved fault tolerance and topology adaptability for scenarios with frequent node fluctuations and unstable links. Future efforts will focus on optimizing communication and encryption mechanisms, introducing compression aggregation and dynamic scheduling, conducting field deployments in multiple scenarios, and integrating adaptive and nonlinear control methods to enhance the system's robustness and real-time performance.

References

[1] Eghmazi A, Ataei M, Landry R J, Chevrette G. Enhancing IoT data security: Using the blockchain to boost data integrity and privacy. IoT, 2024, 5(1):20-34. https://doi.org/10.3390/iot5010002

[2] Shen Y, Shen S, Li Q, Zhou H, Wu Z, Qu Y. Evolutionary privacy-preserving learning strategies for edge-based IoT data sharing schemes. Digital Communications and Networks, 2023, 9(4):906-919. https://doi.org/10.1016/j.dcan.2022.05.004

[3] Akinsanya M O, Ekechi C C, Okeke C D. Security paradigms for iot in telecom networks: conceptual challenges and solution pathways. Engineering Science & Technology Journal, 2024, 5(4):1431-1451. https://doi.org/10.51594/estj.v5i4.1075

[4] Qiu P, Pu Y, Liu Y, Liu W, Yue Y, Zhu X, Ji S. Integer Is Enough: When Vertical Federated Learning Meets Rounding Proceedings of the AAAI Conference on Artificial Intelligence. 2024, 38(13):14704-14712. https://doi.org/10.1609/aaai.v38i13.29388

[5] Dange S, Nitnaware P. Secure Share: Optimal Blockchain Integration in IoT Systems. Journal of Computer Information Systems, 2024, 64(2):265-277. https://doi.org/10.1080/08874417.2023.2193943

[6] Vatambeti R, Krishna E S P, Karthik M G, Damera V K. Securing the medical data using enhanced privacy preserving based blockchain technology in Internet of Things. Cluster Computing, 2024, 27(2):1625-1637. https://doi.org/10.1007/s10586-023-04056-0

[7] Li M, Zhang X, Wang Q, Liu T, Wu R, Wang W, Yu D. Resource-aware federated self-supervised learning with global class representations. Advances in Neural Information Processing Systems, 2024, 37: 10008-10035. https://doi.org/10.52202/079017-0321

[8] Patil R Y. A secure privacy preserving and access control scheme for medical internet of things (MIoT) using attribute-based signcryption. International Journal of Information Technology, 2024, 16(1):181-191. https://doi.org/10.1007/s41870-023-01569-0

[9] Wang X, Jannesari V. Towards a crop pest control system based on the Internet of Things and fuzzy logic. Telecommunication Systems, 2024, 85(4):665-677. https://doi.org/10.1007/s11235-024-01106-9

[10] Zubair M, Iqbal M D A, Shil A, Shil A, Chowdhury M J M, Moni M A, Sarker I H. An improved K-means clustering algorithm towards an efficient data-driven modeling. Annals of Data Science, 2024, 11(5):1525-1544. https://doi.org/10.1007/s40745-022-00428-2

[11] Liu D, Li L. A node clustering algorithm for heterogeneous information networks based on node embeddings. Multimedia Tools and Applications, 2024, 83(2):3745-3766. https://doi.org/10.1007/s11042-023-15245-9

[12] Sinha A, Jana P K. Improved affinity propagation clustering algorithms: A PSO-based approach. Knowledge and Information Systems, 2025, 67(2):1681-1711. https://doi.org/10.1007/s10115-024-02260-x

[13] Abedpour K, Hosseini Shirvani M, Abedpour E. A genetic-based clustering algorithm for efficient resource allocating of IoT applications in layered fog heterogeneous platforms. Cluster Computing, 2024, 27(2):1313-1331. https://doi.org/10.1007/s10586-023-04005-x

[14] Dey A, Bhattacharyya S, Dey S, Platos J, Snasel V. A quantum inspired differential evolution algorithm for automatic clustering of real life datasets. Multimedia tools and applications, 2024, 83(3):8469-8498. https://doi.org/10.1007/s11042-023-15704-3

[15] Wang T, Wu Q, Chen J, Chen F, Xie D, Shen H. Health data security sharing method based on hybrid blockchain[J]. Future Generation Computer Systems, 2024, 153(3):251-261. https://doi.org/10.1016/j.future.2023.11.032

[16] Deabes A G S S, Eid E M, Mansour H A E,. Comparative Analysis of Energy-Efficient Clustering

Algorithms for IoT Networks. Appl. Math, 2024, 18(3):521-530. https://doi.org/10.18576/amis/180304

[17] Rani D, Tripathi S. Design of blockchain-based authentication and key agreement protocol for health data sharing in cooperative hospital network. The Journal of Supercomputing, 2024, 80(2):2681-2717. https://doi.org/10.1007/s11227-023-05577-6

[18] Ji S, Tan Y, Saravirta T, Yang Z, Liu Y, Vasankari L, Walid A. Emerging trends in federated learning: From model fusion to federated x learning. International Journal of Machine Learning and Cybernetics, 2024, 15(9): 3769-3790. https://doi.org/10.1007/s13042-024-02119-1

[19] Zhang X, Zhang R, Chen J, Kim J, Mao Y. Semi-supervised entity alignment with global alignment and local information aggregation. IEEE Transactions on Knowledge and Data Engineering, 2023, 35(10): 10464-10477. https://doi.org/10.1109/TKDE.2023.3238993

[20] Chen Y, Qian W. Multiple reference points-based multi-objective feature selection for multi-label learning. Applied Intelligence, 2024, 54(6): 4952-4978. https://doi.org/10.1007/s10489-024-05387-0

[21] Boulkroune A, Zouari F, Boubellouta A. Adaptive fuzzy control for practical fixed-time synchronization of fractional-order chaotic systems. Journal of Vibration and Control, 2025, 10775463251320258. https://doi.org/10.1177/10775463251320258.

[22] Zouari F, Saad K B, Benrejeb M. Robust neural adaptive control for a class of uncertain nonlinear complex dynamical multivariable systems. International Review on Modelling and Simulations, 2012, 5(5): 2075-2103.