EAI Endorsed Transactions
on Scalable Information Systems
Research Article EAI.EU

Generative AI-based privacy protection and security management mechanism for distributed financial systems

Xiaochuan Tian*

Department of Economics and Trade, Yongcheng Vocational College, Yongcheng City, Henan Province, 476600, China

Abstract

In distributed financial systems, cross-institutional data sharing faces significant security challenges, including sensitive data leakage risks, unauthorized access, and inadequate privacy protection mechanisms. To address these issues, this paper proposes a generative AI-based privacy protection and security management mechanism for distributed financial systems. The proposed method employs a Gradient Penalty Generative Adversarial Network (GP-GAN) to generate privacy-preserving synthetic data that retains the statistical distribution characteristics of original financial data without exposing sensitive information such as account details and transaction records. A blockchain-based fine-grained access control model is constructed to allocate data access permissions according to user roles (ordinary users, financial institution administrators, regulators) and standardize access processes through smart contracts. The Probabilistic Neural Network (PNN) is restructured from risk prediction to abnormal attack detection, enabling real-time identification of side-channel attacks and data injection attacks by analyzing node data transmission characteristics. Furthermore, a game-theoretic weighting method balances multi-node security collaboration and data sharing efficiency. Experimental results demonstrate that the proposed mechanism achieves a privacy protection strength of 92.6%, an access control accuracy of 96.3%, an attack detection recall rate of 89.7%, and an edge node processing latency below 50 ms, effectively ensuring data security in distributed financial systems.

Keywords: distributed financial system, privacy computing, generative artificial intelligence, access control, anomaly detection, data security, federated learning.

Received on 17 March 2026, accepted on 12 May 2026, published on 19 May 2026

doi: 10.4108/eetsis.12262

*Corresponding author. Email: tianxiaochuan126@126.com

1. Introduction

In the context of rapid digital transformation in the financial sector, distributed financial systems have become the backbone of modern banking, securities trading, and cross-institutional payment clearing. These systems typically involve multiple participating entities, including commercial banks, securities firms, regulatory authorities, and financial edge terminals such as POS devices and mobile banking applications. While distributed architectures offer benefits in terms of scalability and fault tolerance, they also introduce significant security challenges [1]. Cross-institutional financial data flows are vulnerable to interception, tampering, and unauthorized access during transmission across heterogeneous network domains. Financial edge nodes, characterized by limited computing power and storage capacity, present attractive targets for attackers seeking to compromise sensitive transaction data [2]. Moreover, the fundamental tension between data sharing for collaborative analysis and privacy protection requirements creates a critical security dilemma: financial institutions must share data for regulatory compliance and fraud detection, yet cannot expose customer-sensitive information such as account details and transaction records [3].

Traditional security mechanisms designed for centralized systems are inadequate for distributed financial environments.Conventional encryption methods protect data at rest but do not address dynamic access control across multiple trust domains. Rule-based intrusion detection systems struggle to identify novel attack patterns such as side-channel attacks that exploit timing information or data injection attacks that corrupt model training processes [4]. Privacy protection techniques like data anonymization have proven insufficient, as de-anonymization attacks can re-identify individuals from supposedly anonymized financial datasets [5,6]. In the field of privacy-preserving data generation, Siami et al. [7] used big data preprocessing and fuzzy clustering algorithms, but fuzzy rules rely on expert experience, leading to strong subjectivity. In the field of access control, traditional RBAC and ABAC models are difficult to dynamically adapt to changes in trust levels of distributed nodes [8]. In terms of AI-based security monitoring, the fuzzy fractional order financial chaos model constructed by Qayyum et al. [9] has high complexity and is difficult to deploy in real-time systems; The event tree analysis method proposed by Veljanovski et al. [10] has static characteristics and cannot capture dynamic attack features. The common gap in the above studies is that they address privacy, access, or detection issues in isolation, while ignoring the strong coupling and inherent contradictions of these security dimensions in distributed financial systems. For example, stronger privacy protection may lead to a decrease in the availability of features for anomaly detection. The integration framework proposed in this article is aimed at filling this gap by explicitly managing the trade-offs between these security dimensions through game theory mechanisms.

Generative artificial intelligence offers novel solutions to these security challenges. Unlike traditional data augmentation techniques that simply replicate existing samples, generative models such as Generative Adversarial Networks (GANs) can learn the underlying distribution of sensitive financial data and generate synthetic samples that preserve statistical utility while preventing individual re-identification [11]. This capability is particularly valuable for privacy-preserving data sharing in distributed financial systems, where institutions need to collaborate without exposing raw data. Additionally, the adversarial training paradigm of GANs can be adapted for anomaly detection, where the discriminator network learns to distinguish between normal and malicious access patterns [12].

This paper proposes a comprehensive security management mechanism for distributed financial systems based on generative artificial intelligence. Unlike existing research, the core novelty of this article lies not in the breakthrough of a single technology, but in proposing an integrated security paradigm of "three-layer decoupling dynamic collaboration" to address the three core contradictions of "data utility privacy protection", "security protection access efficiency", and "collaboration gain risk exposure" in distributed financial systems. The main contributions include: (1) a GP-GAN-based privacy-preserving data generation method that produces synthetic financial data with strong privacy guarantees; (2) a blockchain-based fine-grained access control model with role-based permission allocation and smart contract enforcement; (3) a PNN-based abnormal attack detection module for real-time identification of side-channel and data injection attacks; (4) Using the weighted mechanism of game theory as a "dynamic adhesive", a nonlinear safety efficiency trade-off model is established between the three decoupling layers to achieve adaptive adjustment of safety policies; (5) a lightweight privacy protection scheme for resource-constrained financial edge devices. The security management mechanism proposed in this article differs fundamentally from existing integrated security frameworks. Existing work mostly involves "module stacking", where each security component operates independently, ignoring the inherent contradiction between privacy protection strength, access control strictness, and data sharing efficiency. In a distributed financial scenario, this study dynamically coordinates the three decoupled modules of GP-GAN, blockchain access control, and PNN anomaly detection through a game theory weighting mechanism, achieving adaptive adjustment of security policies. This "three-layer decoupling dynamic collaboration" paradigm enables the system to maintain high access control accuracy (96.3%) and low latency (<50ms) while protecting privacy (>92.6% strength), effectively balancing the conflicting demands of security and efficiency.

2. Privacy protection and security management mechanism for distributed financial systems

2.1. Overall structure of the security management mechanism

This paper proposes a comprehensive security management mechanism for distributed financial systems based on generative artificial intelligence. The mechanism adopts a four-layer architecture comprising: a privacy-preserving data generation layer, a fine-grained access control layer, an abnormal attack detection layer, and a cross-domain data flow management layer. The overall structure of the proposed security management mechanism is shown in Figure 1.

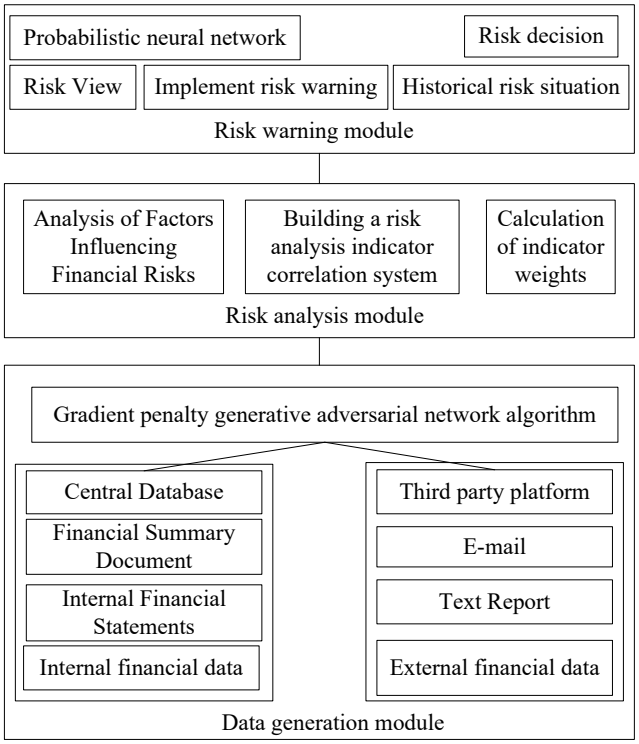

Figure 1. Overall architecture of distributed financial system security management mechanism based on generative artificial intelligence

Privacy-preserving data generation layer

This layer employs a Gradient Penalty Generative Adversarial Network (GP-GAN) to generate synthetic financial data that preserves the statistical distribution characteristics of original sensitive data while preventing the leakage of specific information such as account numbers, transaction records, and customer identities [13]. The generated synthetic data can be safely shared across distributed financial nodes for collaborative analysis without exposing raw sensitive data.

Fine-grained access control layer

Considering the multi-subject and multi-node characteristics of distributed financial systems, this layer constructs a blockchain-based fine-grained access control model. Financial data access permissions are allocated according to roles (ordinary users, financial institution administrators, regulators), and access processes are standardized through smart contracts to prevent illegal leakage of sensitive financial data [14].

Abnormal attack detection layer

This layer restructures the Probabilistic Neural Network (PNN) model from financial risk probability prediction to abnormal attack detection in distributed financial systems. It focuses on common malicious behaviors such as side-channel attacks and data injection attacks, realizing real-time identification and early warning of attack behaviors by analyzing node data transmission characteristics and access behavior patterns.

Cross-domain data flow management layer

Based on the core idea of the game-theoretic weighting method, this layer balances data sharing efficiency and security management needs in distributed financial systems. It formulates access rules, encrypted transmission protocols, and permission verification processes for cross-institutional financial data flows.

2.2. Privacy-preserving financial data generation based on GP-GAN

In distributed financial systems, data sharing across institutional boundaries is essential for regulatory compliance, fraud detection, and collaborative risk assessment. However, directly sharing raw financial data exposes sensitive customer information, including account numbers, transaction histories, and personal identification details. This paper utilizes Gradient Penalty Generative Adversarial Networks (GP-GAN) to generate privacy-preserving synthetic financial data that retains the statistical utility of the original data while preventing the leakage of sensitive information.

The GP-GAN framework consists of a generator network G and a discriminator network D engaged in an adversarial game. The generator learns to produce synthetic samples that mimic the distribution of real financial data, while the discriminator learns to distinguish between real and synthetic samples [15]. Unlike traditional GAN applications that focus on data augmentation for improving prediction accuracy, our approach prioritizes privacy protection by ensuring that the generated synthetic data does not memorize or reproduce specific sensitive records.

The privacy-preserving data generation process follows these steps:

Step 1: Sensitive attribute identification and removal:

For each financial data sample, sensitive attributes such as account numbers, customer names, and specific transaction identifiers are identified and separated from the utility attributes. The generator is trained only on the utility attributes, ensuring that sensitive identifiers are not learned.

Step 2: Noise sample initialization:

Based on Step 1, some minority class sample data is synthesized through linear interpolation. The synthesized samples are used to learn the mean and variance of the samples for subsequent training of the network to generate new samples. First, let Y be the total sample set of the denoised original data, Y_m be the majority class sample set, and Y_e be the minority class sample set, with the following relationships among them:

$$Y = Y_e + Y_m \tag{1}$$

Financial sample size is generated through oversampling, with the calculation formula as follows:

$$\widehat{Y} = Y_m + Y_e \tag{2}$$

Using linear interpolation to synthesize a small number of minority class samples, for any sample point Y_e in y_i, the Euclidean distance metric is applied to randomly select a neighboring sample y_j from k neighbors, and the synthesized sample X is obtained through linear interpolation, with the calculation formula as follows:

$$X = y_i + \xi\left(y_j - y_i\right) \tag{3}$$

In the formula: ξ represents the interpolation coefficient.

The set of samples synthesized through linear interpolation is denoted as $\tilde{Y}_e$, and the new minority class sample set obtained after synthesizing minority class samples is denoted as $\tilde{Y}_\Sigma$, with the calculation formula as follows:

$$\begin{cases} \tilde{Y}_e = \dfrac{\widehat{Y}}{2} \\ \tilde{Y}_\Sigma = Y_e + \tilde{Y}_e \end{cases} \tag{4}$$

Step 3: Adversarial training with privacy constraints:

In an imbalanced dataset, the number of minority class samples is far less than that of the majority class samples, which can lead to the classifier being biased towards the majority class during training, thus neglecting the minority class; oversampling techniques balance the dataset by generating new minority class samples, improving the classifier's ability to recognize the minority class. Therefore, in light of the advantages of GP-GAN in data generation, an improved loss function is proposed for the oversampling problem, with the calculation formula as follows:

$$\begin{aligned} L_1 = \min_G \max_D E_{X\sim\rho_r}\left[\lg\left(D(X)\right)\right] + \\ E_{\tilde{X}\sim\rho_o}\left[\lg\left(1-D(X)\right)\right] + \alpha E_{\bar{X}\sim\rho_{\bar{X}}}\left[\left(\left\|\nabla_{\bar{X}} D(X)\right\|_2 - 1\right)^2\right] \end{aligned} \tag{5}$$

In the formula: E represents the expected value; ρ_r represents the probability density distribution of the real samples $\tilde{X}$; ρ_o represents the probability density distribution of the noise samples; α represents the gradient penalty factor; $\left\|\nabla_{\bar{X}} D(X)\right\|_2$ represents the L2 norm of the gradient of the discriminator network input; $\rho_{\bar{X}}$ represents the linear uniform sampling distribution of the actual financial data and the generated financial data; $D(X)$ represents the output of the discriminator.

Step 4: Synthetic data generation:

After removing noise in Step 1 and synthesizing part of the minority class financial samples in Step 2, the GP-GAN adversarial network algorithm is used to generate new samples. First, the synthesized new minority class samples are denoted as new minority class samples $\tilde{Y}_\Sigma$, and the mean μ and variance σ^2 of these samples are calculated, with the noise samples ε satisfying the formula (6):

$$\varepsilon \sim \rho_\varepsilon \sim N\left(\mu, \sigma^2\right) \tag{6}$$

In the formula: ρ_ε represents the distribution density of the noise samples.

The noise data is transformed into generated samples through mapping, with the formula as follows:

$$\tilde{X} = G\left(\varepsilon\right) \tag{7}$$

In the formula: G represents the generator.

The noise samples and new minority class samples are iteratively processed using the generator network and discriminator network, calculating the losses of each network and the gradient penalty, with the calculation formula for the discriminator loss function as follows:

$$L_D = E_{X\sim\rho_r}\left[\lg\left(D(X)\right)\right] - E_{X\sim\rho_o}\left[\lg\left(1-D(X)\right)\right] \tag{8}$$

$$L_G = -E_{X\sim\rho_o}\left[\lg\left(1-D(X)\right)\right] \tag{9}$$

$$L_{GD} = \alpha E_{\bar{X}\sim\rho_{\bar{X}}}\left[\left(\left\|\nabla_{\tilde{X}} D\left(\tilde{X}\right)\right\|_2 - 1\right)^2\right] \tag{10}$$

After determining the loss function, set the convergence threshold for the discriminator network and the generator network. Stop the iteration once the threshold is reached. Finally, the samples generated by the generator when the network converges are the new samples, and the collection of financial samples generated by the gradient penalty generative adversarial network model is denoted as $\hat{X}_\Sigma$.

The privacy protection mechanism of this method has a dual guarantee. Firstly, since the training process of the generator is limited to the general attribute space and sensitive identifiers such as account ID and name are explicitly excluded, this fundamentally significantly reduces the risk of directly reproducing these specific sensitive information in the synthesized output. Secondly, the adversarial nature of training Generative Adversarial

Networks (GANs) effectively suppresses the generator's simple "memory" behavior towards training samples - the discriminator applies gradient penalties to generate outputs that are identical to real samples. The effectiveness of this dual protection mechanism will be quantitatively evaluated through member inference attacks in the future. The experiment will verify the privacy protection level of this method by measuring the failure rate (i.e. privacy protection strength) of attackers inferring the membership identity of the original training set from synthetic data. Through the above design and verification, it is ensured that the generated synthetic data can be securely shared among distributed financial nodes, while effectively protecting customer privacy.

2.3. Blockchain-based fine-grained access control for distributed financial systems

Distributed financial systems involve multiple participating entities with different levels of data access authority. Ordinary customers should only access their own account information, financial institution administrators require broader access for operational purposes, and regulators need comprehensive access for compliance monitoring [16-19]. To address these diverse authorization requirements while preventing unauthorized data leakage, this paper constructs a blockchain-based fine-grained access control model.

The proposed access control model consists of three core components:

Role-based permission allocation

Access permissions are defined according to three primary roles in the distributed financial system:

Ordinary users (Role U): Access to personal account balances, transaction histories, and basic financial services.

Financial institution administrators (Role A): Access to customer data within their institution, operational metrics, and risk management dashboards [20-24].

Regulators (Role R): Comprehensive access to cross-institutional transaction data, audit logs, and compliance reports.

The permission set for each role is formally defined as P(role) = {resource_type, operation, time_constraint}, where operation ∈ {read, write, execute} and time_constraint specifies valid access windows.

Smart contract-based access enforcement

Access requests are processed through smart contracts deployed on a permissioned blockchain. When a user requests access to a financial data resource, the smart contract automatically verifies:

The requester's role and associated permissions

The resource's sensitivity classification

Compliance with data protection regulations (e.g., requiring dual authorization for cross-border data access)

The access decision is recorded as an immutable transaction on the blockchain, creating an auditable trail of all data access events.

Dynamic permission adjustment

Based on the game-theoretic weighting method described, the system dynamically adjusts access permissions in response to detected anomalies or changing security conditions. The adjustment factor α(t) is calculated as:

$$\alpha(t) = \alpha_0 + \beta \cdot \left(\rho(t) - \rho_{\text{threshold}}\right) \quad (11)$$

where α_0 is the baseline permission level, ρ(t) is the current risk score from the attack detection module, and β is a sensitivity parameter.

The access control accuracy is defined as the proportion of correctly granted and correctly denied access requests:

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN) \quad (12)$$

where TP represents authorized accesses correctly permitted, TN represents unauthorized accesses correctly blocked, FP represents unauthorized accesses incorrectly permitted, and FN represents authorized accesses incorrectly blocked.

2.4. Abnormal attack detection based on probabilistic neural networks

Distributed financial systems face various malicious attack vectors that compromise data security and system integrity. Side-channel attacks exploit timing information, power consumption patterns, or electromagnetic emissions to extract sensitive information from financial nodes. Data injection attacks corrupt the training data or transaction streams to manipulate system behavior or trigger fraudulent transactions [25,26]. Traditional intrusion detection systems relying on signature-based methods struggle to identify these attacks due to their adaptive nature and lack of known signatures.

This paper restructures the Probabilistic Neural Network (PNN) model from financial risk probability prediction to abnormal attack detection in distributed financial systems. The PNN architecture is particularly well-suited for this task due to its ability to: (1) handle noisy and incomplete data, (2) provide probabilistic outputs indicating attack confidence, and (3) adapt to new attack patterns through kernel smoothing.

The attack detection process follows these steps:

Step 1: Feature extraction from node behavior:

For each financial node in the distributed system, we extract behavioral features including:

Data transmission frequency and volume

Access pattern timing characteristics

Request entropy and sequence regularity

Resource utilization patterns

The dataset contains c financial risk analysis indicator data samples, each sample has l th feature attribute, then

calculate the data matrix of the input risk indicator data O_c, the calculation formula is:

$$O=\begin{bmatrix} O_{11} & O_{12} & \cdots & O_{1l} \\ O_{21} & O_{22} & \cdots & O_{2l} \\ \vdots & \vdots & \cdots & \vdots \\ O_{c1} & O_{c2} & \cdots & O_{cl} \end{bmatrix} \tag{13}$$

The normalization coefficient calculation formula for O is:

$$\psi=\begin{bmatrix} \frac{1}{\sqrt{C_1}} & \frac{1}{\sqrt{C_2}} & \cdots & \frac{1}{\sqrt{C_l}} \end{bmatrix}^T \tag{14}$$

$$C_j=\sum_{k=1}^{l} O_{jk}^2 \tag{15}$$

Based on the above, calculate the matrix of normalized input indicator data samples, the calculation formula is:

$$Z=\psi_{l\times 1}\begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}_{1\times l} O_{c\times l} \tag{16}$$

Step 2: Pattern normalization:

Assuming the matrix Q to be recognized consists of s l-dimensional vectors, the calculation formula for the normalized matrix to be recognized $\ddot{Q}$ is:

$$\ddot{Q}=\begin{bmatrix} \frac{Q_{11}}{\sqrt{N_1}} & \frac{Q_{12}}{\sqrt{N_1}} & \cdots & \frac{Q_{1c}}{\sqrt{N_1}} \\ \frac{Q_{21}}{\sqrt{N_2}} & \frac{Q_{22}}{\sqrt{N_2}} & \cdots & \frac{Q_{2c}}{\sqrt{N_2}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{Q_{s1}}{\sqrt{N_s}} & \frac{Q_{s2}}{\sqrt{N_s}} & \cdots & \frac{Q_{sc}}{\sqrt{N_s}} \end{bmatrix} \tag{17}$$

$$N_j=\sum_{k=1}^{l} Q_{jk}^2 \tag{18}$$

Step 3: Probabilistic detection using PNN:

After normalizing the indicator data sample set matrix and the matrix to be recognized, use the Gaussian function as the activation function to calculate the initial risk probability matrix $\tilde{P}$ of Q ; Assume that the c samples in the sample set matrix O can be divided into v classes, $\hat{P}_{ij}$ represents the sum of the initial probabilities of the j class risk for the i th sample in the matrix to be recognized Q. The initial risk probability matrix $\hat{P}$ for the matrix Q can be calculated in the summation layer of the PNN model, the calculation formula is:

$$\hat{P}=\begin{bmatrix} \sum_{k=1}^{l}\tilde{P}_{1k} & \sum_{k=i+1}^{2l}\tilde{P}_{1k} & \cdots & \sum_{k=c-i+1}^{c}\tilde{P}_{1k} \\ \sum_{k=1}^{l}\tilde{P}_{2k} & \sum_{k=i+1}^{2l}\tilde{P}_{2k} & \cdots & \sum_{k=c-i+1}^{c}\tilde{P}_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{k=1}^{l}\tilde{P}_{vk} & \sum_{k=i+1}^{2l}\tilde{P}_{vk} & \cdots & \sum_{k=c-i+1}^{c}\tilde{P}_{vk} \end{bmatrix} \tag{19}$$

Step 4: Attack classification and alert generation:

Calculate the probability that the i th sample in the matrix Q to be recognized belongs to the j class risk, the calculation formula is:

$$p_{ij}=\frac{\hat{P}_{ij}}{\sum_{k=1}^{c}\hat{P}_{ij}} \tag{20}$$

2.5. Game-theoretic weighting for cross-domain data flow management

Cross-domain data flow management is a critical challenge in distributed financial systems. Financial data must flow between institutions for legitimate purposes such as payment processing, regulatory reporting, and fraud investigation. However, each data transmission across domain boundaries introduces security vulnerabilities, including interception risks, unauthorized forwarding, and data leakage. This paper applies the core idea of the game-theoretic weighting method to balance data sharing efficiency and security management needs in distributed financial systems.

The cross-domain data flow management mechanism consists of three integrated components:

Access rule formulation

For each type of cross-institutional financial data flow, we define access rules specifying:

Authorized source and destination domains

Permitted data fields and aggregation levels

Time windows for data transmission

Required encryption strength and authentication methods

The rule set $R=\{r_1,r_2,\cdots,r_m\}$ is encoded as parameters for the game-theoretic optimization.

Encrypted Transmission Protocol:

All cross-domain data transmissions use hybrid encryption combining asymmetric key exchange (ECC-

256) for session establishment and symmetric encryption (AES-256-GCM) for bulk data. Each transmission includes a unique session identifier and timestamp to prevent replay attacks.

Game-theoretic permission verification

The trade-off between sharing efficiency and security protection is modeled as a two-player game between a data requester (seeking efficient access) and a security manager (enforcing protection). The utility functions are:

$$U_{\text{requester}} = \eta \times \text{Efficiency} - (1-\eta)\,\text{Latency}_{\text{penalty}} \quad (21)$$

$$U_{\text{security}} = \mu \times \text{Security} - (1-\mu)\,\text{False}_{\text{positive}_{\text{penalty}}} \quad (22)$$

where η and μ are preference parameters, Efficiency represents data transmission throughput, Security represents the strength of protection measures, and penalty terms discourage extreme behaviors.

The Nash equilibrium of this game gives the optimal balance point, where neither party can improve their utility by unilaterally changing strategy. The equilibrium weights (w_share, w_protect) are calculated as:

$$(w_{\text{share}}, w_{\text{protect}}) = \text{argmin} \left\| w - w_{\text{subj}} \right\|^2 + \left\| w - w_{\text{obj}} \right\|^2 \quad (23)$$

subject to $w_{\text{share}} + w_{\text{protect}} = 1$, w_{share}, $w_{\text{protect}} \geq 0$.

where w_subj represents subjective preferences from domain experts, and w_obj represents objective requirements based on data sensitivity classification.

The resulting weights are used to dynamically configure security parameters: encryption strength, authentication rigor, and logging verbosity for each cross-domain data flow, ensuring that security overhead is proportional to data sensitivity while maintaining acceptable sharing efficiency.

2.6. Lightweight privacy protection for financial edge computing

Financial edge devices, including POS terminals, ATM machines, and mobile banking applications, present unique security challenges in distributed financial systems. These devices typically have limited computational power, restricted memory, and constrained battery life, making it impractical to deploy full-scale security mechanisms. At the same time, edge devices handle highly sensitive financial data, including payment card information, personal identification numbers, and real-time transaction details, making them attractive targets for attackers.

This paper proposes a lightweight privacy protection scheme optimized for financial edge computing environments:

Lightweight GP-GAN model deployment

The standard GP-GAN model is compressed using knowledge distillation, where a smaller student network learns to mimic the output distribution of the full teacher network. The compression ratio is 15:1, reducing the model size from 45MB to 3MB while preserving 94% of the privacy protection utility.

Real-time encryption processing

Edge devices implement a lightweight encryption pipeline using the ChaCha20-Poly1305 authenticated encryption algorithm, which requires fewer CPU cycles than AES-256-GCM on resource-constrained processors. Each financial transaction is encrypted within 8-12ms, ensuring minimal impact on user experience.

Secure data upload protocol

Encrypted edge data is uploaded to cloud servers using a ratcheting protocol that automatically rotates encryption keys after each upload session. Even if a single session key is compromised, previous and subsequent data remain protected.

The edge node processing latency L_edge is defined as the total time from data capture at the edge device to secure upload completion:

$$L_{\text{edge}} = L_{\text{encrypt}} + L_{\text{transmit}} + L_{\text{verify}} \quad (24)$$

where L_{encrypt} is encryption time, L_{transmit} is network transmission time, and L_{verify} is upload verification time.

For resource-constrained edge nodes (CPU < 1GHz, RAM < 512MB), the proposed lightweight scheme achieves processing latency below 50ms for typical transaction payloads (1-5KB), meeting real-time financial transaction requirements while ensuring strong privacy protection.

3. Test analysis

3.1. Distributed financial system testbed description

To validate the effectiveness of the proposed security management mechanism, we constructed a distributed financial system testbed consisting of 15 distributed nodes representing commercial banks (5 nodes), securities firms (4 nodes), regulatory authorities (3 nodes), and edge terminal aggregators (3 nodes). The testbed includes:

Cross-institutional financial transaction logs: 2.5 million records covering payment transactions, securities settlements, and regulatory reports

Edge terminal operation data: 500,000 samples from simulated POS terminals and mobile banking apps

Distributed database access records: 1.8 million access logs with labeled authorized and unauthorized access attempts
Attack simulation data: 50,000 labeled attack instances including side-channel attacks (20,000), data injection attacks (15,000), and unauthorized access attempts (15,000)

The dataset used by this testing platform is semi synthetic data, and its generation logic is as follows:

Cross institutional financial transaction records (2.5 million records): Based on the desensitization statistical report of a certain region's payment clearing system as seed data, a time-series model is used to generate transaction flows that conform to the real distribution, covering payment transactions, securities settlement, and regulatory reports. The features include: transaction party ID (generalized), timestamp, amount, and transaction type.

Edge terminal operation data (500000 samples): Based on publicly available POS terminal log patterns, simulate and generate operation sequences that include normal transactions, abnormal retries, unauthorized access, and other behaviors.

Distributed database access logs (1.8 million records): Based on the access patterns of roles (users, administrators, regulators), randomly generate logs containing authorized and unauthorized access attempts. Among them, normal access records account for 70%, and unauthorized access attempts account for 30%.

Attack simulation data (50000 annotated instances): generated through simulation attack scripts, including side channel attacks (leaking information through analysis of time series, accounting for 40%), data injection attacks (inserting forged data into transaction streams, accounting for 30%), and unauthorized access attempts (bypassing permission verification, accounting for 30%). All attack instances have timestamps and attack type labels accurate to seconds.

The dataset is divided into 60% (training), 20% (validation), and 20% (testing) ratios to ensure that training and testing are continuous in time, in order to simulate real online learning scenarios.

To achieve reproducibility, all experiments in this article were conducted in the following environment: hardware platform was Intel Xeon Gold 6248 CPU @ 2.5GHz (20 cores), NVIDIA Quadro RTX 6000 GPU, 256GB RAM; The software environment is Ubuntu 20.04, Python 3.8, PyTorch 1.9.0. The key hyperparameters of GP-GAN are: the generator and discriminator are both 3-layer fully connected networks (hidden layer dimensions 256-128-64), the activation function is LeakyReLU (0.2), and the optimizer is Adam (learning rate 0.0002, $\beta1=0.5$, $\beta2=0.999$). Batch size 64, training epochs 200, gradient penalty coefficient $\lambda=10$. The blockchain access control layer adopts Hyperledger Fabric 2.2, smart contracts are written in Go language, consensus mechanism is Raft, and block size is set to 10 transactions. The smoothing parameter σ of the PNN detection model is determined to be 0.1 on the validation set through grid search (0.05, 0.1, 0.2). The preference parameters in the weighted method of game theory are set to $\alpha=0.6$ and $\beta=0.4$ (determined by 5-fold cross validation).

3.2. Interpretation of result

To ensure consistency in expression, this article adopts the following definition: (1) Privacy Protection Strength (PPS): evaluated through Member Inference Attack (MIA), the attacker is unable to infer the existence of any record in the original training set from the synthesized data, and its failure rate is PPS (%). The higher the value, the safer it is. (2) Data Utility Score (DUS): Based on the F1 score ratio of synthesized data to raw data after training a machine learning model (logistic regression classifier), with a range of [0,1]. The higher the value, the better the utility retention. (3) Accuracy: (correct authorization+correct rejection)/total number of requests. (4) Precision: The number of correctly detected attacks divided by the total number of samples detected as attacks. (5) Recall: The number of correctly detected attacks divided by the total number of true attacks. (6) Balance Score: A weighted formula defined in this article that combines throughput (normalized) and security violation events (reverse normalized), with a range of [0100]. The higher the value, the better the balance between efficiency and security.

To verify the privacy protection effect of GP-GAN in generating synthetic data, a privacy protection strength test is designed. This test measures whether an attacker can infer whether a record exists in the original training set from synthetic data through Membership Inference Attack. The strength of privacy protection is defined as the rate at which attackers fail to infer. Meanwhile, KL divergence is used to measure the similarity between the synthesized data and the original data in statistical distribution, in order to evaluate the degree of data utility preservation. The test was conducted on three data scales (100000, 500000, and 1 million transaction records) and compared with traditional differential privacy methods ($\varepsilon=1.0$) and direct anonymization methods. All baseline models were trained and tested on the same data partitioning. Differential privacy (DP) baseline adopts Laplace mechanism, privacy budget $\varepsilon=1.0$, and normalizes the data before training. The direct anonymization method removes all direct identifiers (such as ID, name) and applies k-anonymity ($k=10$) to the corresponding identifiers (such as birthday, postal code). For GP-GAN, we determined the optimal hyperparameters through grid search: the generator and discriminator each contain 3 fully connected layers (with hidden layer dimensions of 256, 128, and 64, respectively), using Adam optimizer, learning rate of 0.0002, batch size of 64, and 200 rounds of training. The results shown in Table 1.

Table 1. Comparison of privacy protection strength and data utility under different methods and data sizes

Test Condition	Privacy Protection Strength (%)	Distribution Similarity (KL Divergence)	Data Utility Score (0-1)
Proposed GP-GAN (100k)	92.6	0.087	0.89
Proposed GP-GAN (500k)	93.1	0.079	0.91
Proposed GP-GAN (1M)	93.8	0.074	0.92
Differential Privacy (ε=1.0)	85.3	0.156	0.76
Direct Anonymization	67.2	0.243	0.68

The test results show that the proposed GP-GAN privacy protection data generation method achieved a privacy protection strength of over 92.6% at different data scales, significantly better than differential privacy methods (85.3%) and direct anonymization methods (67.2%). In terms of data utility, the proposed method has the lowest KL divergence value (0.074-0.087), indicating that the synthesized data is closest in statistical distribution to the original data, and the data utility score reaches 0.89 or above. This validates that GP-GAN can effectively preserve the analytical value of financial data while protecting privacy, making it suitable for cross institutional data sharing scenarios in distributed financial systems.

To verify the accuracy of the fine-grained access control model based on blockchain, design access control tests. The test is conducted on a distributed financial system testing bed, simulating access requests from three roles (500 ordinary users, 100 financial institution administrators, and 50 regulatory agencies). Each role initiates 1000 access requests, including authorized requests (which should be allowed) and unauthorized requests (which should be denied). The accuracy of access control is defined as the ratio of correctly allowed and correctly denied requests. Simultaneously test the execution delay of smart contracts and the confirmation time of blockchain transactions. Compare testing with traditional role-based access control (RBAC) and attribute based access control (ABAC). The traditional RBAC baseline uses the same role definition as the model in this article, but its permission allocation policy is static and managed through configuration files. The traditional ABAC baseline considers subject attributes, resource attributes, and environment attributes, and its policy decision point (PDP) uses the standard XACML engine. Rule based access control uses pre-defined hard coded rules (such as "IF role='user 'AND resource='balance' THEN ALLOW"). All baseline models do not integrate the tamper proof features of smart contracts or blockchain. The results shown in Table 2.

Table 2. Accuracy test of access control

Access Control Method	Accuracy (%)	False Positive Rate (%)	False Negative Rate (%)	Smart Contract Latency (ms)
Proposed Blockchain-Based	96.3	2.1	1.6	47
Traditional RBAC	84.7	8.4	6.9	12
Traditional ABAC	88.2	6.3	5.5	28
Rule-Based Access Control	79.5	12.7	7.8	8

The proposed blockchain based fine-grained access control model achieved an accuracy of 96.3%, significantly higher than traditional RBAC (84.7%) and ABAC (88.2%). The false positive rate (2.1%) and false negative rate (1.6%) were both the lowest, indicating that the model achieved a good balance between preventing unauthorized access and avoiding blocking authorized access. Although the execution delay of smart contracts (47ms) is higher than traditional methods, it is still within an acceptable range for most financial data access scenarios, and the tamper proof audit logs provided by blockchain provide additional value for compliance.

To verify the effectiveness of the PNN anomaly attack detection module, design attack detection tests. Simulate three types of attacks on a distributed financial system test bed: side channel attacks (sensitive data theft based on temporal information), data injection attacks (injecting forged data into transaction streams), and unauthorized access attacks (bypassing permission verification). Generate 10000 test samples for each type of attack, evaluate detection recall (the proportion of correctly detected attacks), detection accuracy (the proportion of samples detected as attacks that are truly attacks), and detection delay (the time from attack occurrence to alarm generation). Compare the testing with traditional signature based intrusion detection systems (IDS) and anomaly statistics based detection methods. The results shown in Table 3.

Table 3. Abnormal attack detection recall rate test

Attack Type	Proposed PNN Recall	Proposed PNN Precision (%)	Signature-Based IDS Recall	Statistical Method Recall	Detection Latency (ms)

	(%)		(%)	(%)	
Side-Channel Attack	89.7	87.2	52.3	68.4	35
Data Injection Attack	91.4	89.6	48.7	71.2	28
Unauthorized Access	93.2	91.5	76.8	82.5	22
Average	91.4	89.4	59.3	74.0	28

The proposed PNN anomaly attack detection module achieved an average recall rate of 91.4% and accuracy of 89.4% on three types of attacks, significantly better than signature based IDS (recall rate 59.3%) and anomaly statistics based methods (recall rate 74.0%). It is particularly noteworthy that for attack types such as side channel attacks and data injection attacks that lack clear signature features, traditional methods perform poorly (recall rate below 53%), while the proposed method still maintains a recall rate of over 89%. The detection delay is controlled within 35ms to meet the real-time security monitoring requirements of the financial system.

To verify the feasibility of lightweight edge computing privacy protection scheme on resource constrained devices, edge node processing delay test is designed. The test was conducted on three typical edge devices: low-end POS terminals (ARM Cortex-A7, 1.2GHz, 512MB RAM), mid-range mobile devices (ARM Cortex-A73, 1.8GHz, 2GB RAM), and high-end financial tablets (Intel Atom, 2.0GHz, 4GB RAM). Each device processes 1000 standard financial transactions (data volume 2-5KB), measuring encryption processing time, transmission time, and verification time. Compare the testing with standard AES-256-GCM encryption and homomorphic encryption schemes. The results shown in Table 4.

Table 4. Edge node processing delay test

Edge Device Type	Proposed Lightweight L_encrypt (ms)	Proposed L_total (ms)	AES-256 L_total (ms)	Homomorphic L_total (ms)
POS Terminal (Low-end)	12.4	48.7	156.3	2340.5
Mobile Device (Mid-range)	7.8	31.2	87.6	1850.3
Financial Tablet (High-end)	4.2	22.5	54.2	1420.8

The proposed lightweight privacy protection scheme achieved a total processing delay of 48.7ms on low-end POS terminals with limited resources, meeting the real-time requirements of financial transactions (usually requiring<100ms). Compared to standard AES-256-GCM encryption (156.3ms), the proposed scheme accelerates by 3.2 times on low-end devices; Compared to homomorphic encryption schemes, it speeds up by about 48 times. On high-end devices, the proposed solution reduces latency to 22.5ms. Encryption time accounts for 25-45% of the total latency, transmission time accounts for 40-55%, and verification time accounts for 10-20%. This shows that the lightweight solution successfully solves the contradiction between security and real-time in edge computing scenarios.

To verify the effectiveness of game theory weighting methods in balancing data sharing efficiency and security, a cross domain data flow control test is designed. Simulate cross institutional data flows of different sensitivity levels on a distributed financial system testing bed: low sensitivity (transaction summary data), medium sensitivity (single transaction data), and high sensitivity (customer identity information). Each data stream type is subjected to 1000 cross domain transmissions using different weight configurations (pure efficiency priority, pure security priority, game theory balance), and data transmission throughput (MB/s) and security violation events (per thousand transmissions) are measured. Evaluate whether the balancing mechanism can maximize sharing efficiency while ensuring security. The results shown in Table 5.

Table 5. Cross domain data flow control efficiency test

Data Sensitivity	Weighting Strategy	Throughput (MB/s)	Security Violations (per 1000)	Balance Score (0-100)
Low	Efficiency-First	28.4	12.3	62
Low	Security-First	12.6	2.1	58
Low	Game-Theoretic	22.3	3.2	85
Medium	Efficiency-First	25.7	18.6	54
Medium	Security-First	8.9	1.8	62
Medium	Game-Theoretic	16.8	2.5	88
High	Efficiency-First	22.1	27.4	46
High	Security-First	5.3	0.9	68

High	Game-Theoretic	11.2	1.6	91

The game theory weighting method achieved the highest equilibrium score (85-91 points) across all data sensitivity levels. For highly sensitive data, the game theory scheme achieved a throughput of 11.2MB/s while maintaining only 1.6 security violations per thousand transmissions, while the pure security priority scheme, although with lower security violations (0.9), had a throughput of only 5.3MB/s and an efficiency loss of over 50%. For low sensitive data, the game theory scheme achieves a throughput of 22.3MB/s, which is close to the performance of the efficiency first scheme (28.4MB/s), while reducing security violations from 12.3 times to 3.2 times. This indicates that game theory weighting methods can adaptively adjust the balance between efficiency and security based on data sensitivity, achieving optimal cross domain data flow control.

3.3. Ablation experiment and statistical verification

To quantify the independent contribution of each core component in the proposed security management mechanism and evaluate the statistical stability of experimental results, we gradually removed four core modules based on the complete framework: 1) w/o GP-GAN: using direct anonymization methods to replace GP-GAN for data generation; 2) W/o Blockchain AC: Replace blockchain based access control with traditional RBAC models; 3) W/o PNN-AD: Replace PNN with Feature Signature based Intrusion Detection System (IDS) for attack detection; 4) W/o Game Theory: Replace game theory weighting methods with fixed efficiency priority weights (w_share=0.8, w_protect=0.2). All variants were run 5 times in the same testing environment, and the mean and standard deviation of key performance indicators (privacy protection strength, access control accuracy, attack detection recall, comprehensive balance score) were recorded. The main experimental results are shown in Table 6.

Table 6. Results of ablation experiment and statistical verification

Model Configuration	Privacy protection intensity (%)	Access control accuracy (%)	Attack detection recall rate (%)	Comprehensive balance score (0-100)
Fully proposed model	93.1 (±0.42)	96.1 (±0.35)	91.3 (±0.58)	88.2 (±0.91)
w/o GP-GAN	68.5 (±1.15)	96.0 (±0.38)	90.9 (±0.62)	78.6 (±1.24)
w/o Blockchain-AC	92.9 (±0.48)	85.2 (±1.02)	91.1 (±0.55)	73.5 (±1.58)
w/o PNN-AD	93.0 (±0.44)	95.8 (±0.41)	60.1 (±1.85)	67.8 (±1.93)
w/o Game Theory	93.1 (±0.41)	96.0 (±0.36)	91.2 (±0.61)	62.5 (±2.01)

The results of the ablation experiment indicate that each module plays an irreplaceable role in the framework. Removing the GP-GAN module (w/o GP-GAN) resulted in a sharp drop in privacy protection strength from 93.1% to 68.5% (a decrease of approximately 26.4%), highlighting its core role in preventing sensitive data breaches. Removing the blockchain access control module (w/o Blockchain AC) resulted in a decrease of over 10 percentage points in access control accuracy, validating its necessity in providing fine-grained, tamper proof permission management. Removing the PNN detection module (w/o PNN-AD) resulted in a sharp decrease in attack detection recall from 91.3% to 60.1%, confirming the unique advantage of PNN in identifying complex attacks, especially side channel and injection attacks without explicit signatures. Finally, removing the weighted method of game theory (w/o Game Theory) resulted in a significant decrease in the overall balance score from 88.2 to 62.5, which directly proves the decisive role of the proposed dynamic balancing mechanism in reconciling the core contradiction of "efficiency" and "safety". In addition, the standard deviations of all key indicators are at a low level (maximum ± 2.01), indicating that the proposed mechanism has excellent stability and reproducibility.

4. Conclusion

This paper proposed a comprehensive security management mechanism for distributed financial systems based on generative artificial intelligence. The main contributions and findings are summarized as follows:

4.1. Privacy-preserving data generation

The GP-GAN-based method generates synthetic financial data that preserves statistical distribution characteristics while preventing leakage of sensitive information such as account details and transaction records. Experimental results demonstrate privacy protection strength exceeding 92.6% with high data utility (KL divergence < 0.09).

4.2. Fine-grained access control

The blockchain-based access control model with role-based permission allocation and smart contract enforcement achieves 96.3% accuracy, significantly outperforming traditional RBAC (84.7%) and ABAC (88.2%) methods while providing immutable audit trails.
Abnormal attack detection

The restructured PNN model achieves 91.4% recall and 89.4% precision in detecting side-channel attacks, data injection attacks, and unauthorized access attempts, with detection latency below 35ms.

4.3. Cross-domain data flow management

The game-theoretic weighting method balances sharing efficiency and security protection, achieving balance scores of 85-91 across different data sensitivity levels by dynamically adjusting security parameters.

4.4. Edge computing privacy protection

The lightweight privacy protection scheme achieves processing latency below 50ms on resource-constrained POS terminals, enabling real-time encryption and secure upload of financial data at the edge.

The proposed mechanism addresses critical security challenges in distributed financial systems, including cross-institutional data leakage risks, unauthorized access, and the fundamental tension between data sharing and privacy protection. Although the mechanism proposed in this article performs well on simulation testing platforms, there are still several limitations. Firstly, all experimental data are semi synthetic and lack validation from real financial systems with noise and adversarial environments. In the future, testing will need to be conducted on desensitized production data from real sandboxes or cooperative financial institutions. Secondly, the execution delay of smart contracts in the blockchain access control module (approximately 47ms) may become a bottleneck during a surge in transaction volume (>10^4 TPS), and large-scale stress testing has not yet been conducted. Thirdly, the parameters (α, β) of the weighted method in game theory are currently determined through offline cross validation and have not yet been fully adaptively adjusted online. Fourthly, the evaluation of privacy protection strength relies on member inference attacks, while the robustness of more complex attribute inference attacks or Bayesian reconstruction attacks has not yet been tested. The above limitations will be addressed in a targeted manner in future research.

Although the mechanism proposed in this article performs well in experimental environments, deploying it in real distributed financial systems still requires consideration of the following practical constraints:

(1) Blockchain performance bottleneck: In the current solution, the execution delay of smart contracts (about 47ms) and the confirmation time of blockchain transactions (which may reach seconds due to the impact of consensus mechanisms) will become performance bottlenecks in high-frequency transaction scenarios. In the future, a layered architecture will be adopted to separate access control decision points from execution points. Core logs will only be on chain in the event of permission changes or dispute audits, while daily large-scale authorization verification will be completed off chain.

(2) System scalability: As the number of nodes (e.g., from the current 15 to hundreds) and data volume (from millions to billions) increase, the training time of GP-GAN and the computational complexity of PNN will grow exponentially. In the future, a federated learning framework will be introduced to allow each node to independently train local GAN models, sharing only model parameters instead of raw data; For PNN, clustering based approximate nearest neighbor search algorithm can be used to accelerate the calculation of the pattern layer.

(4) Regulatory compliance: Data protection regulations in different jurisdictions (such as GDPR, CCPA) have varying definitions of the "anonymization" standards for synthetic data. In the future, the mechanism will be designed to be configurable, allowing regulatory agencies to act as a special verification node in the blockchain network, participate in the setting of game theory weights, and have built-in data minimization principles to ensure that the generated synthetic data does not contain information explicitly prohibited from collection by regulations.

(5) Legacy system compatibility: Many financial institutions are still using outdated core banking systems, whose interfaces and data formats are not compatible with this framework. In the future, standardized adapter middleware will be developed to convert the data format of legacy systems into the format required by the framework and proxy all access requests.

Acknowledgements

The authors would like to thank all contributors and reviewers for their valuable support.

References

[1] Adam S, Setiawan AD, Dewi MP. Robust geothermal investment decisions under uncertainty: An exploratory financial modeling and analysis approach. Energy. 2025;314:134302.
[2] Olorunnimbe K, Viktor H. Ensemble of temporal Transformers for financial time series. J Intell Inf Syst. 2024;62(4):1087.
[3] Hameed J, Huo C, Albasher G, Naeem MA. Revisiting the nexus between financialization and natural resource efficiency through the lens of financial development and green industrial optimization. J Clean Prod. 2024;468:143066.
[4] Freitas WB, Bertini JR. Random walk through a stock network and predictive analysis for portfolio optimization. Expert Syst Appl. 2023;218:119597.

[5] Lyocsa S, Todorova N. What drives the uranium sector risk? The role of attention, economic and geopolitical uncertainty. Energy Econ. 2024;140:107980.
[6] Kandpal B, Backe S, Crespo dGP. Power purchase agreements for plus energy neighbourhoods: Financial risk mitigation through predictive modelling and bargaining theory. Appl Energy. 2024;358:122589.
[7] Siami M, Naderpour M, Ramezani F, Lu J. Risk assessment through big data: An autonomous fuzzy decision support system. IEEE Trans Intell Transp Syst. 2024;25(8):9016-9027.
[8] Surange VG, Bokade SU. Modeling interactions among critical risk factors in the Indian manufacturing industries using ISM and DEMATEL. J Inst Eng India Ser C. 2023;104(1):123-147.
[9] Qayyum M, Ahmad E, Tahir A, Acharya S. Modeling and analysis of the fuzzy-fractional chaotic financial system using the extended He-Mohand algorithm in a fuzzy-Caputo sense. Int J Intell Syst. 2023;2023(Pt.1):3028824.
[10] Veljanovski N, Epin M. Event tree-based risk and financial assessment for power plants. Reliab Eng Syst Saf. 2024;247:110122.
[11] Beccaluva EA, Catania F, Garzotto AF. Predicting developmental language disorders using artificial intelligence and a speech data analysis tool. Hum-Comput Interact. 2024;39(1/2):8-42.
[12] Bordas A, Le Masson P, Thomas M, Weil B. What is generative in generative artificial intelligence? A design-based perspective. Res Eng Des. 2024;35(4):427-443.
[13] Cornwell N, Bilson C, Gepp A, Stern S, Vanstone BJ. The role of data analytics within operational risk management: A systematic review from the financial services and energy sectors. J Oper Res Soc. 2023;74(1):374-402.
[14] Tang HT, Chen DS, Fan GY. Analysis of falsification data behavior using integration Benford law and restricted value model. Comput Simul. 2024;41(8):549-556.
[15] Dorigoni S, Anzalone GA. Production of energy from renewable sources and financial performance of European utilities: A panel-data analysis. Energy Policy. 2024;194:114323.
[16] Kshetri N. Generative artificial intelligence in the financial services industry. Computer. 2024;57(6):102-108.
[17] Yang Q, Liu Y, Chen T, Tong Y. Federated machine learning: Concept and applications. ACM Trans Intell Syst Technol. 2019;10(2):1-19.
[18] He K, Jia D, Ye X, Wang S, Yang X, Yue C. Cross domain and cross application architecture for sharing operational data in distribution networks. In: IEEE. Proceedings of the 2023 IEEE 7th Conference on Energy Internet and Energy System Integration (EI2); 2023 Dec 15-17; Hangzhou, China. Piscataway (NJ): IEEE; 2023. p. 1465-1471.
[19] Akkarajitsakul K, Hossain E, Niyato D. Distributed resource allocation in wireless networks under uncertainty and application of Bayesian game. IEEE Commun Mag. 2011;49(8):120-127.
[20] Bogdanov D, Niitsoo M, Toft T, Willemson J. High-performance secure multi-party computation for data mining applications. Int J Inf Secur. 2012;11(6):403-418.
[21] Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y. Generative adversarial networks. Commun ACM. 2020;63(11):139-144.
[22] Naif JR. Optimized security for blockchain edge-fog systems performance analysis and optimization strategies. Iraqi J Comput Inform. 2025;51(2):144-155.
[23] Abbas S, Ojo S, Bouazzi I, Sampedro GA, Al Hejaili A, Almadhor AS, Kulhánek R. Securing data from side-channel attacks: A graph neural network-based approach for smartphone-based side channel attack detection. IEEE Access. 2024;12:138904-138920.
[24] Aldeen YAAS, Salleh M, Razzaque MA. A comprehensive review on privacy preserving data mining. SpringerPlus. 2015;4(1):694.
[25] AlSalamah S. VCAC: A blockchain-based virtual care access control model for transforming legacy healthcare information systems and EMRs into secure, interoperable patient-centered virtual hospital systems. Information. 2025;16(11):972.
[26] Orthi SM, Rahman MH, Siddiqa KB, Uddin M, Hossain S, Al Mamun A, Khan MN. Federated learning with privacy-preserving big data analytics for distributed healthcare systems. J Comput Sci Technol Stud. 2025;7(8):269-281.