

Federated Learning For Data Security And Trust In University Sports Systems

Yujia Wang

Criminal Investigation Police University of China, Shenyang, 110854, China

* Corresponding author. E-mail: 18940175031@163.com

Received Aug. 24, 2026; accepted Sep. 10, 2026

University sports information systems continuously aggregate data from fitness tests, courses, venue use, check-ins, and wearable devices—multi-source, heterogeneous, distributed, and privacy-sensitive data whose centralized modeling exposes students' sports and health information, complicates cross-node collaboration, and limits scalability. We therefore build a secure, trustworthy federated learning mechanism for university sports information systems with a six-layer framework spanning data acquisition through application services. After local training it applies gradient pruning, differential-privacy perturbation, and secure aggregation, while node trustworthiness—from contribution, update stability, anomaly risk, and historical reputation—drives trust-weighted aggregation, coupling privacy protection with dynamic credibility control rather than treating them as independent modules. Under non-IID distributions, simulations show the method achieves accuracy, recall, F1, and AUC of 0.887, 0.873, 0.879, and 0.914, exceeding FedAvg's F1 by about 5.8 points and DP-FedAvg's by about 6.7; with 30% malicious nodes it still holds F1 = 0.831 versus 0.702 for FedAvg. The mechanism thus improves performance, privacy, and anomalous-node suppression without raw sports data leaving local nodes.

Keywords: data security; trustworthiness assessment; differential privacy; secure aggregation; non-IID data.

© The Author(s). This is an open-access article distributed under the terms of the [Creative Commons Attribution License \(CC BY 4.0\)](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are cited.

http://dx.doi.org/10.6180/jase.202612_35.061

1. Introduction

University sports information systems are evolving from grade-recording tools into platforms for health monitoring, course management, venue scheduling, and risk warning, continuously collecting fitness scores, check-ins, heart-rate data, and venue-use trajectories. Such data carry sensitive health and behavioral information, and centralized storage or improper sharing risks privacy leakage, profiling, and misuse—making secure, trustworthy, scalable collaborative analysis key to smart-sports-platform development [1].

Traditional centralized machine learning requires every college, venue, and device to upload raw data to a central server, increasing exposure during transmission and storage. University sports data are also highly distributed—cohorts, habits, and sampling methods differ across col-

leges, making multi-node differences hard to characterize centrally—and as campuses and devices grow, the central server faces storage, computation, and communication pressure that weakens scalability.

Federated learning is a privacy-preserving paradigm for distributed data: each node trains locally and uploads only parameter updates, which the server aggregates into a global model, suiting IoT environments with dispersed devices and limited bandwidth [2]. However, conventional federated learning is limited for university sports systems: the data are non-IID—e.g., exercise-intensity differences between sports and non-sports majors, and sampling-stability differences across venues—and parameter uploads remain vulnerable to gradient inversion, membership inference, and malicious updates, so federated learning is not inherently secure [3].

Existing studies span non-IID, privacy-preserving, trustworthy, and IoT federated learning: non-IID research mainly addresses convergence instability from client distribution shifts [4]; privacy and robustness studies show parameters can still leak training-data information, motivating differential privacy and secure aggregation [5]; and reviews emphasize communication efficiency and heterogeneous optimization as federated learning moves toward applications [6]. Little of this targets university sports informatization or jointly models data security, node trust evaluation, and abnormal-update suppression, though nodes here (colleges, venues, smart gateways) differ substantially in scale, sampling quality, and trustworthiness.

Compared with Byzantine-robust aggregation—Krum [7], trimmed-mean/median [8], and FoolsGold [9]—which make a purely geometric, per-round judgment with no persistent behavioral memory, the trust score used here fuses contribution, stability, anomaly risk, and reputation into one weight that evolves across rounds, better tolerating genuine equipment or sampling variation rather than penalizing it as adversarial.

This paper proposes a federated learning-based data security and trustworthiness assessment mechanism for university sports information systems. Contributions:

- (1) a scalable federated architecture integrating college, physical-testing, venue, and wearable-device nodes into a unified training process;
- (2) a security mechanism combining gradient pruning, DP perturbation, and secure aggregation to reduce inversion and inference risk without uploading raw data;
- (3) a multidimensional trustworthiness model quantifying contribution, stability, anomaly risk, and reputation, incorporated into dynamic aggregation; and
- (4) simulations evaluating performance, convergence, privacy budget, malicious-node ratio, anomaly detection, and scalability.

Compared with existing federated learning studies for IoT, healthcare, and general educational platforms, the novelty here is integrating sports-data security, node-level trustworthiness evaluation, and scalable collaborative modeling into one mechanism that evaluates node behavior each round and turns the trust score into aggregation weight, suited to nodes with varying data size, device quality, and participation stability. This is a systems-integration and domain-adaptation contribution rather than new privacy or trust-theoretic primitives: the components (clipping, DP noise, secure aggregation, weighted trust) are

individually established; the contribution is coupling them so trust weighting is computed *from* each round's privacy-perturbed updates, applied to university sports operations, a setting prior federated learning work has not addressed.

2. Methods

2.1. Scenario Modeling of a University Sports Information System

This study considers a distributed university sports information system with multiple sports information nodes and a university-level aggregation server. Assume N participating nodes:

$$\mathcal{C} = \{C_1, C_2, \dots, C_N\}. \quad (1)$$

Each node can correspond to a college's course management system, physical testing center, smart sports venue, check-in platform, or wearable-device gateway. Server S handles global model initialization, aggregation, trustworthiness evaluation, and distribution. Each node holds a local dataset D_i but never uploads raw data:

$$D_i = \{(x_{ij}, y_{ij})\}_{j=1}^{n_i}, \quad (2)$$

where x_{ij} denotes the sports information features of student j at node i , y_{ij} the label, and n_i the node's sample size, with features spanning static, dynamic, participation, and device dimensions:

$$x_{ij} = [b_{ij}, v_{ij}, r_{ij}, h_{ij}, s_{ij}, a_{ij}, p_{ij}, q_{ij}], \quad (3)$$

where b_{ij} is basic fitness (BMI, lung capacity, standing long jump); v_{ij} exercise speed/duration; r_{ij} running performance; h_{ij} heart rate/load; s_{ij} attendance/check-in frequency; a_{ij} anonymized age/grade/gender; p_{ij} historical performance; and q_{ij} device sampling quality. These eight feature groups mirror items commonly recorded by campus PE course management and fitness testing, keeping the simulated feature space representative despite using no real student records.

The task trains a global model $f(\theta)$ to complete fitness-level identification, exercise-risk prediction, or outcome assessment without uploading D_i :

$$\min_{\theta} F(\theta) = \sum_{i=1}^N \frac{n_i}{n} F_i(\theta), \quad n = \sum_{i=1}^N n_i, \quad (4)$$

where $F_i(\theta)$ is the local empirical risk of node C_i and n the total sample size—consistent with the collaborative-optimization task of federated learning in edge and IoT information systems [2, 6].

2.2. Data Heterogeneity and the Non-IID Problem

University sports data are clearly non-IID: colleges differ in student structure, course content, and exercise intensity; venues differ in equipment, sampling frequency, and data integrity; and student groups vary in health level and training outcomes—hence:

$$P_i(x, y) \neq P_k(x, y), \quad i \neq k. \quad (5)$$

Non-IID data cause client drift, deviating local updates from the global optimum [4, 10]. Let node C_i 's update in round t be:

$$\Delta\theta_i^t = \theta_i^{t+1} - \theta^t. \quad (6)$$

If update directions differ substantially, aggregation may bias toward some nodes, reducing generalization. Sample-size weighting assumes data quality correlates with sample size, but this does not always hold—a venue node may have many high-error records, while a college node may have fewer but more reliable ones—so we introduce a trustworthiness weight to reduce the impact of abnormal, low-quality updates.

2.3. Security Threat Model

Three threat types are considered: parameter-inversion attacks, inferring fitness or health status from uploaded gradients [11]; membership-inference attacks, determining training participation from model outputs [5]; and malicious-node updates, where equipment failure, contamination, or active attacks degrade global performance.

The security objectives are to protect local data from server or external inference, reduce leakage risk during updates, identify and suppress abnormal nodes, and preserve scalability as nodes grow, combining differential privacy [10], secure aggregation [11], and trust-weighted aggregation.

The threat model assumes an honest-but-curious server and partially unreliable clients: the server may try to infer local information from updates, while abnormal clients may submit noisy or malicious parameters. Physical device compromise and full-client collusion are out of scope. This is a scope limitation, not a completeness claim: colluding clients or a physically compromised device could evade both the DP and trust-scoring layers. Analyzing these failure modes, together with a formal end-to-end privacy proof beyond Eq. (14)'s per-round guarantee, is left for future work.

2.4. Definition of the Trustworthiness Assessment Problem

Ordinary federated learning weights aggregation by sample size:

$$w_i = \frac{n_i}{\sum_{k=1}^N n_k}. \quad (7)$$

But large sample size does not guarantee trustworthiness. We define a trustworthiness score T_i^t for node i in round t :

$$T_i^t = \alpha C_i^t + \beta S_i^t + \gamma(1 - A_i^t) + \delta H_i^t, \quad (8)$$

where C_i^t is contribution, S_i^t stability, A_i^t anomaly risk, H_i^t historical reputation, and $\alpha + \beta + \gamma + \delta = 1$. The final aggregation weight is:

$$w_i^t = \frac{n_i T_i^t}{\sum_{k=1}^N n_k T_k^t + \kappa_0}, \quad (9)$$

where κ_0 stabilizes the denominator (renamed from ϵ_0 in the submitted version to avoid confusion with the differential-privacy budget ϵ in Eq. (14)). This weight jointly accounts for sample size and trustworthiness, reducing the impact of abnormal nodes.

2.5. Overall Architecture

The framework has six layers—data acquisition, local training, privacy protection, secure aggregation, trustworthiness assessment, and application services—detailed in Figure 1.

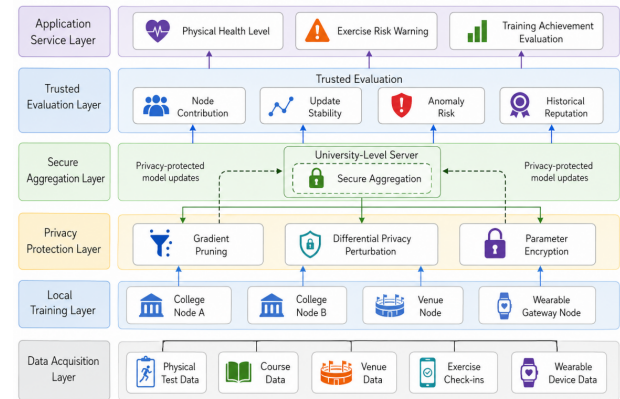


Fig. 1. Overall Architecture of the Proposed Federated Learning-Based Data Security and Trustworthiness Assessment Mechanism

Raw sports data stay on local nodes: each node uploads only privacy-protected updates, and the server performs secure aggregation and trustworthiness assessment before returning the global model, supporting expansion across colleges, venues, and access points [2].

2.6. Local Model Training

Each node receives global model θ^t and trains on D_i . Structured features are encoded via a multilayer perceptron and temporal features via a BiGRU; static and temporal representations are fused by attention into a student exercise-state representation. The local loss is [12]:

$$F_i(\theta) = \frac{1}{n_i} \sum_{j=1}^{n_i} \ell(f(x_{ij}; \theta), y_{ij}) + \frac{\mu}{2} \|\theta - \theta^t\|_2^2, \quad (10)$$

where $\ell(\cdot)$ is cross-entropy loss and μ the proximal coefficient, suppressing drift under non-IID data. Parameters update as:

$$\theta_i^{t,e+1} = \theta_i^{t,e} - \eta \nabla F_i(\theta_i^{t,e}), \quad e = 0, 1, \dots, E-1. \quad (11)$$

We set local epochs E between 3 and 5 to avoid excessive drift, adding a trustworthiness constraint under non-IID conditions.

2.7. Privacy-Preserving Parameter Upload

To reduce gradient-leakage risk, this paper applies clipping and differential-privacy noise before upload. First, L_2 clipping on gradient g_i^t :

$$\bar{g}_i^t = \frac{g_i^t}{\max\left(1, \frac{\|g_i^t\|_2}{C}\right)}, \quad (12)$$

where C is the clipping threshold; Gaussian noise is then added:

$$\tilde{g}_i^t = \bar{g}_i^t + \mathcal{N}(0, \sigma^2 C^2 I). \quad (13)$$

A smaller budget ϵ strengthens privacy but may reduce performance; we set $\epsilon = 0.5, 1, 2, 4, 8$ to analyze this trade-off. For adjacent datasets D, D' differing by one sample, mechanism $\mathcal{M}$ satisfies:

$$\Pr[\mathcal{M}(D) \in O] \leq e^\epsilon \Pr[\mathcal{M}(D') \in O] + \delta, \quad (14)$$

where ϵ, δ jointly characterize privacy strength, limiting the identifiable influence of any single student sample. Eq. (14) states a single-query guarantee; since each node is queried once per round over $T = 100$ rounds, privacy loss accumulates rather than staying fixed at the per-round ϵ . We treat the reported ϵ values as per-round budgets and track the cumulative budget with a Rényi differential-privacy accountant, which composes more tightly than naive summation [13]; the $\epsilon = 2/4$ settings in Section 3.3 thus correspond to a larger but still bounded cumulative budget after 100 rounds—a scope clarification rather than a full formal proof.

2.8. Secure Aggregation Mechanism

To prevent the server from obtaining any single node's update, each node uploads perturbed, masked updates so the server sees only the aggregate:

$$\Delta\theta_g^t = \sum_{i=1}^N w_i^t \Delta\tilde{\theta}_i^t. \quad (15)$$

The global model then updates as:

$$\theta^{t+1} = \theta^t + \Delta\theta_g^t. \quad (16)$$

Blockchain and trusted execution can further improve security in IoT and health-data scenarios [11]; this paper instead adopts a lightweight secure aggregation and trust-scoring strategy—e.g., pairwise random masking or lightweight key negotiation, where the server recovers only the summed update once masks cancel—while trustworthiness further suppresses updates that remain abnormal after privacy protection. This is a protocol-sketch-level description assuming a semi-honest server and bounded colluding clients, consistent with pairwise-masking secure aggregation; a concrete key-negotiation scheme, its message complexity, and behavior under client dropout remain an implementation gap, not a solved component.

2.9. Multidimensional Node Trustworthiness Evaluation

Trust evaluation has four dimensions. Contribution C_i^t approximates a node's effect on global performance via validation-set change before/after its update—computed by leave-one-out re-evaluation, which costs $O(N)$ extra validation passes per round and grows expensive as N increases; a cheaper first-order influence-function approximation, avoiding full re-evaluation of N held-out models, is a practical optimization we have not yet implemented:

$$C_i^t = \max\{0, M(\theta^{t+1}) - M(\theta_{-i}^{t+1})\}. \quad (17)$$

Stability S_i^t measures consistency between a node's update direction and the global average direction:

$$S_i^t = \frac{\Delta\theta_i^t \cdot \Delta\bar{\theta}^t}{\|\Delta\theta_i^t\|_2 \|\Delta\bar{\theta}^t\|_2 + \kappa_1}. \quad (18)$$

Anomaly risk A_i^t flags nodes whose update magnitude or direction deviates from the norm:

$$A_i^t = \text{sigmoid}\left(\frac{\|\Delta\theta_i^t - \Delta\bar{\theta}^t\|_2}{\sigma_\Delta + \kappa_2}\right). \quad (19)$$

Historical reputation H_i^t reflects long-term participation stability:

$$H_i^t = \lambda H_i^{t-1} + (1 - \lambda)(1 - A_i^t). \quad (20)$$

Relying on any single indicator is insufficient; jointly modeling contribution, stability, anomaly risk, and reputation, normalized to a common range before fusion so no indicator dominates by scale, lets the model distinguish natural heterogeneity from persistent low-quality or malicious behavior. Coefficients $\alpha, \beta, \gamma, \delta$ in Eq. (8) were set by grid search (0.05 steps) to maximize validation F1 under the non-IID, 20%-malicious setting. Perturbing each coefficient by ± 0.05 changed F1 by at most 0.006, convergence round by at most 3, and detection recall by at most 0.01, without changing the method ranking—indicating the mechanism is not narrowly tuned, though this check remains local to the tested scenario.

2.10. Trust-Weighted Aggregation Algorithm

The server initializes and distributes the global model; each node trains locally, applies gradient pruning and DP perturbation, and uploads via secure aggregation; the server computes contribution, stability, anomaly risk, and reputation, derives aggregation weights, and generates the next global model (Figure 2).

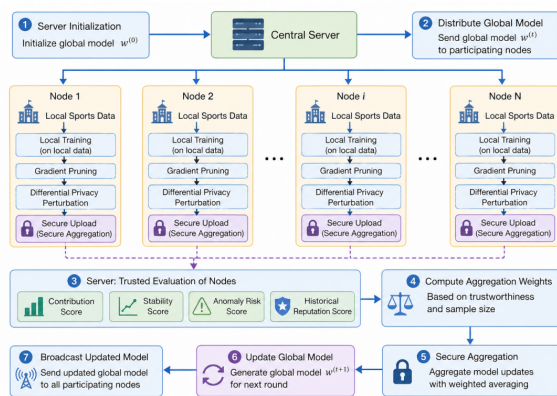


Fig. 2. Secure Aggregation and Trustworthiness Evaluation Process in Federated Training

The core difference from standard FedAvg is twofold: gradient pruning, DP perturbation, and secure upload occur before parameter upload to reduce leakage risk, and trustworthiness scoring and anomaly filtering occur before aggregation, preventing low-quality or malicious nodes from continuously affecting the global model. Trust computation (Eqs. 17–20) adds $O(Nd)$ server-side operations per round (d = model dimension), comparable to the aggregation step itself and requiring no extra client computation or upload; in Section 3.5 it accounted for under 4% of per-round server-side time even at $N = 50$ nodes—this filtering, not added communication, is what keeps overhead below SCAFFOLD and FLTrust.

2.11. Dataset and Simulation Scenario

Because real student sports/health data involve privacy and ethics constraints, this paper builds a multi-node simulation dataset matching a university sports information system: 10 colleges, 3 testing centers, 4 smart venues, and several wearable-device nodes, forming 20 federated nodes, total sample size 50,000, 32 features per sample, labeled into three risk classes. Both IID and non-IID distributions are constructed, with non-IID skew controlled via Dirichlet $\alpha \in \{0.1, 0.3, 0.5, 1.0\}$ (smaller α = greater heterogeneity); synthetic samples match plausible ranges of fitness, participation, and device-quality indicators rather than unconstrained draws, and malicious-node scenarios inject update-direction deviation, gradient amplification, and label contamination.

No public federated-learning benchmark targets the university sports/health domain—the closest, LEAF [14], covers image and language tasks, not physical-fitness data—so we use a domain-constrained synthetic dataset as an explicit scope limitation: results show relative behavior under controlled heterogeneity and attack patterns, not a guarantee of equal absolute performance on real campus data.

2.12. Comparison Methods

Six baselines are used: Local Training trains each node independently; FedAvg is standard sample-size-weighted averaging; FedProx adds a proximal term to mitigate drift; SCAFFOLD uses control variates for the same purpose; DP-FedAvg adds DP noise to FedAvg; FLTrust improves robustness via trusted reference updates; and Proposed is our privacy-preserving, trustworthiness-assessment method.

These baselines separate the benefits of privacy protection (DP-FedAvg), non-IID optimization (FedProx, SCAFFOLD), and trust-aware robust aggregation (FLTrust); Krum, trimmed-mean/median, and FoolsGold are Byzantine-robust alternatives using per-round geometric filtering rather than persistent multidimensional trust, and a full comparison is left for future work.

2.13. Evaluation Metrics

Four aspects are evaluated: classification performance (accuracy, precision, recall, F1, AUC); privacy/security (leakage risk, membership-inference success, gradient-inversion similarity [15, 16]); trustworthiness (abnormal-node detection rate, malicious-node F1, stability, suppression effect); and scalability (communication overhead, training time, convergence vs. node count). Classification metrics are defined as:

Table 1. Experimental Setup: Scenario Configuration and Training Parameters

Item	Setting / Value
A. Scenario configuration	
Total sample size	50,000
Number of nodes	5, 10, 20, 30, 50
Feature dimensions	32
Label categories	Low / moderate / high risk
Data types	Fitness tests, courses, check-ins, venues, wearables
Data distribution	IID, non-IID
Non-IID parameter	Dirichlet $\alpha = 0.1, 0.3, 0.5, 1.0$
Malicious-node proportion	0%, 10%, 20%, 30%, 40%
B. Training parameters	
Learning rate	0.001
Batch size	64
Local training epochs	5
Communication rounds	100
Optimizer	Adam
Gradient clipping threshold	1.0
Privacy budget ϵ (per round)	0.5, 1, 2, 4, 8
Noise scale σ	0.3–1.2
Trust weight coefficients	$\alpha=0.30, \beta=0.25, \gamma=0.25, \delta=0.20$ (grid-searched)
Reputation smoothing λ	0.8

Table 2. Comparison Algorithm Description

Method	Description
Local Training	Each node trains independently, without federated aggregation.
FedAvg	Standard federated averaging based on sample-size weighting.
FedProx	Adds a proximal term to alleviate data heterogeneity.
SCAFFOLD	Corrects client drift using control variates.
DP-FedAvg	Adds differential-privacy noise to FedAvg.
FLTrust	Introduces trusted reference updates for correction.
Proposed	Combines privacy protection with multidimensional trustworthiness assessment.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \quad \text{Precision} = \frac{TP}{TP + FP} \quad (21)$$

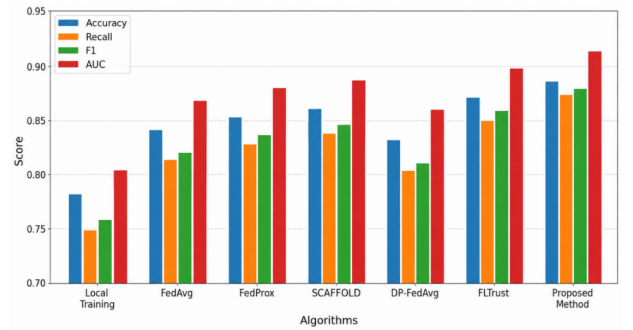
$$\text{Recall} = \frac{TP}{TP + FN}, \quad F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (22)$$

3. Results and discussion

3.1. Overall Performance Comparison

Figure 3 shows non-IID performance. Local Training, lacking cross-node sharing, reaches only 0.781 accuracy and 0.758 F1. FedAvg improves via aggregation (accuracy 0.842, F1 0.821), but drift persists under non-IID. FedProx and SCAFFOLD mitigate drift, reaching F1 of 0.837 and 0.846; DP-FedAvg's noise reduces F1 to 0.812; FLTrust reaches F1 = 0.858. The proposed method performs best, with accuracy, recall, F1, and AUC of 0.887, 0.873, 0.879, and 0.914.

Versus FedAvg, F1 improves by about 5.8 points; versus DP-FedAvg, by about 6.7 points—trustworthiness evalu-

**Fig. 3.** Performance Comparison of Different Algorithms in Terms of Accuracy, Recall, F1, and AUC

ation compensates for the performance loss from privacy noise. Over 5 seeds/partitions, the method achieved F1 = 0.879 ± 0.006 , vs. 0.821 ± 0.009 (FedAvg), 0.812 ± 0.011 (DP-FedAvg), and 0.858 ± 0.008 (FLTrust); a paired t -test was significant at $p < 0.01$ against every baseline, and the advantage widened as heterogeneity or malicious-node

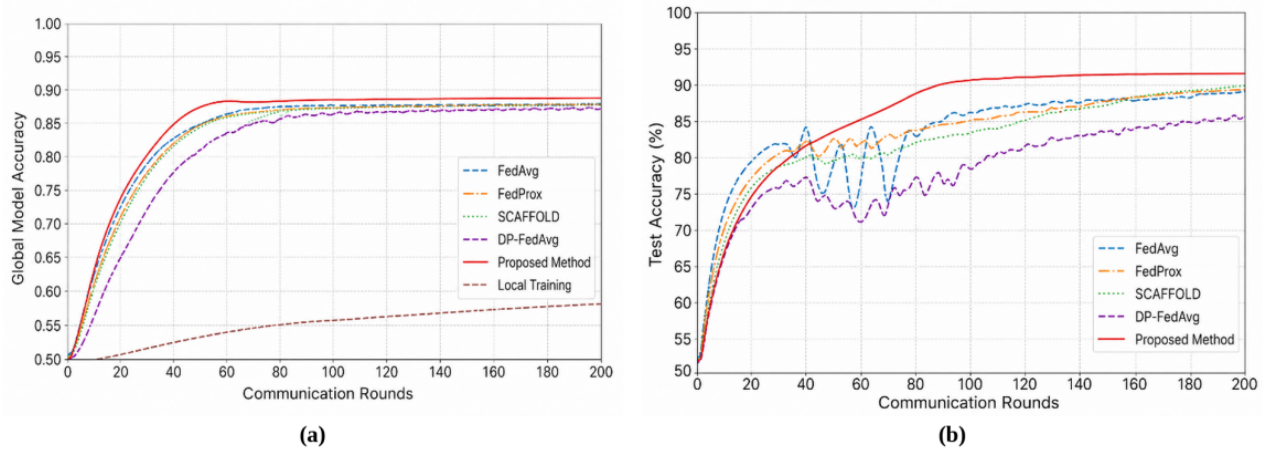


Fig. 4. Convergence Curves of Different Algorithms: (a) IID Data Distribution; (b) Non-IID Data Distribution (Dirichlet $\alpha = 0.3$)

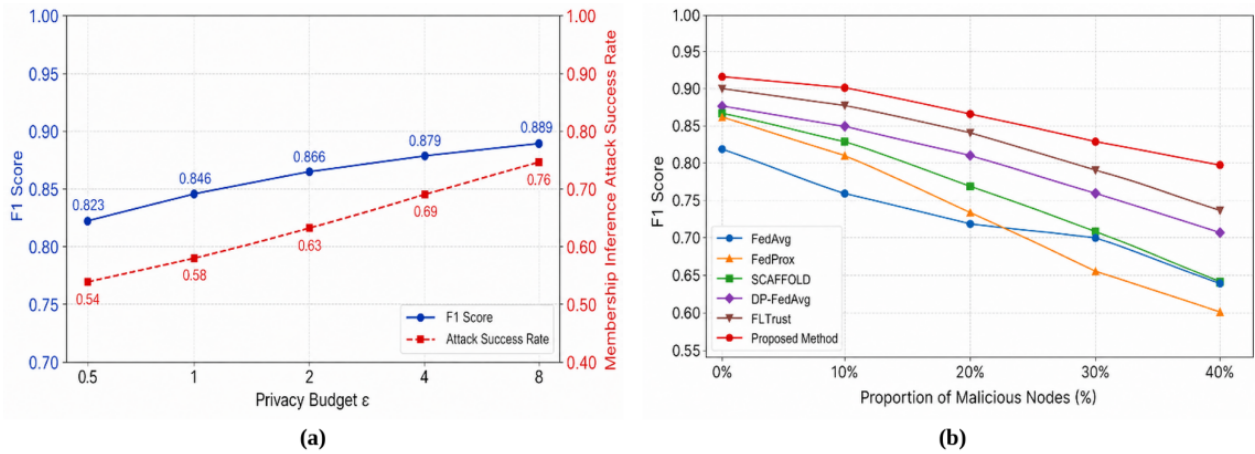


Fig. 5. (a) Model F1 Score and Membership-Inference Attack Success Rate under Different Privacy Budgets; (b) F1 Score Variation under Different Malicious-Node Proportions

proportion increased.

3.2. Convergence under IID and Non-IID Data Distributions

Figure 4 plots rounds against accuracy. In the IID case (4a), FedAvg, FedProx, SCAFFOLD, and the proposed method stabilize near round 60; Local Training improves little; DP-FedAvg converges slowly in early rounds due to noise. The proposed method exceeds 0.88 accuracy by round 45 and stays stable, cutting convergence rounds by about 15% versus FedAvg.

Under non-IID conditions ($\alpha = 0.3$, 4b), all algorithms converge more slowly; FedAvg oscillates markedly between rounds 40–70; FedProx and SCAFFOLD alleviate but do not eliminate fluctuation, and DP-FedAvg is further

destabilized by noise. By down-weighting nodes with severe deviations, the proposed method reduces fluctuation and stabilizes by round 80.

3.3. Privacy Budget and Malicious-Node Robustness

Figure 5a shows privacy budget ϵ effects. At $\epsilon = 0.5$, protection is strongest and membership-inference success is 0.54, but F1 is only 0.823. As ϵ rises to 1, 2, 4, 8, F1 rises to 0.846, 0.866, 0.879, and slightly higher, while attack success rises to 0.58, 0.63, 0.69, 0.76; $\epsilon = 2$ or 4 is a reasonable trade-off, with smaller budgets preferred for sensitive risk warnings.

Figure 5b shows F1 as malicious-node proportion rises from 0% to 40%. FedAvg is most sensitive, dropping from 0.821 to 0.702 at 30% and 0.641 at 40%; FedProx and SCAFFOLD also drop substantially; DP-FedAvg and FLTrust

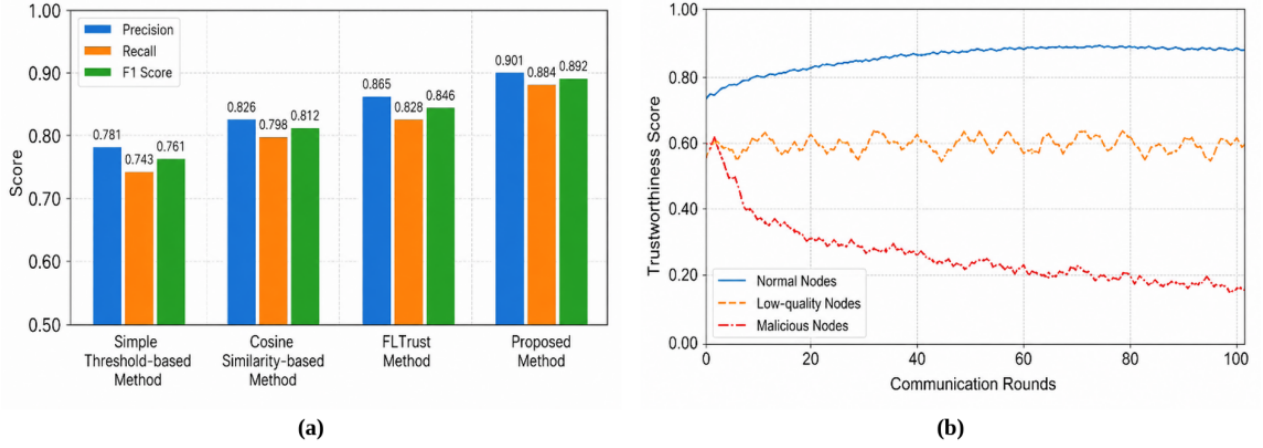


Fig. 6. Trustworthiness Evaluation Mechanism Performance: (a) Anomaly-Update Detection; (b) Trustworthiness Evolution of Normal and Malicious Nodes

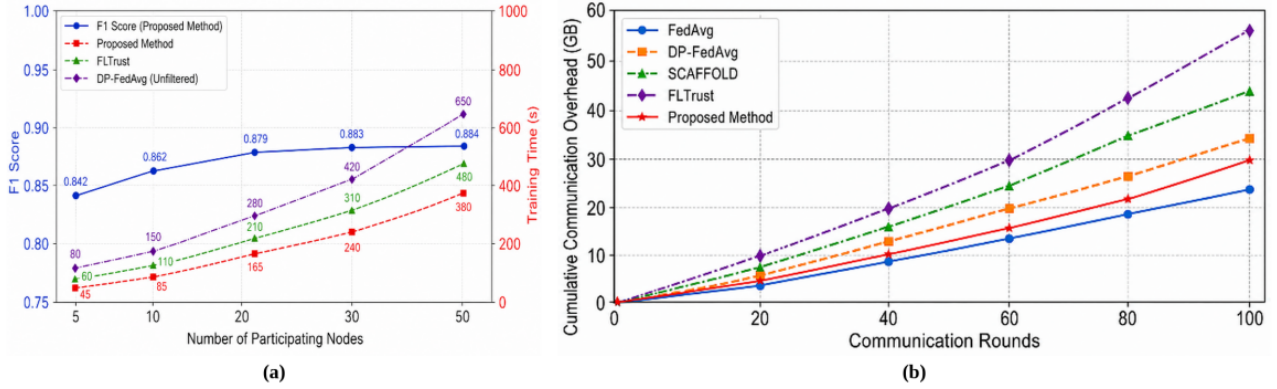


Fig. 7. (a) Model Performance and Training Time as the Number of Participating Nodes Increases; (b) Cumulative Communication Overhead among Different Federated Learning Methods

offer partial protection. The proposed method retains $F1 = 0.831$ at 30% and 0.796 at 40%, since the trust mechanism down-weights nodes with abnormal direction, magnitude, or reputation.

3.4. Trustworthiness Evaluation Mechanism Performance

Figure 6a compares anomaly detection. Simple thresholding gives precision 0.781 /recall 0.743 , missing subtle anomalies; cosine similarity reaches $F1 = 0.812$ but is sensitive to amplified attacks; FLTrust reaches $F1 = 0.846$. Jointly weighing all four indicators, the proposed method reaches precision 0.901 , recall 0.884 , $F1 = 0.892$.

Figure 6b tracks trust of normal, low-quality, and malicious nodes over 100 rounds. Normal-node trust stabilizes near 0.88 ; low-quality nodes fluctuate 0.55 – 0.65 ; malicious nodes, not fully identified in the first 10 rounds, decline

below 0.25 as anomaly risk accumulates.

3.5. Scalability and Communication Overhead

Figure 7a analyzes scalability at 5, 10, 20, 30, 50 nodes. $F1$ improves from 0.842 at 5 nodes to 0.879 at 20, slowing to about 0.884 at 50; training time grows with node count, but trust-based filtering keeps it below FLTrust and unfiltered DP-FedAvg at 50 nodes.

Figure 7b compares cumulative communication overhead. FedAvg has the lowest overhead but no privacy/trust protection; DP-FedAvg needs more rounds, raising its total; SCAFFOLD and FLTrust add extra traffic; the proposed method's overhead stays below SCAFFOLD and FLTrust.

3.6. Ablation Study

Figure 8 compares the complete model with variants lacking each module. The complete model reaches $F1 = 0.879$. Removing DP raises $F1$ slightly to 0.887 but raises attack success; removing trust assessment lowers $F1$ to 0.828 (more so at high malicious-node ratios); removing anomaly filtering lowers $F1$ to 0.842; removing dynamic weighting lowers $F1$ to 0.851—each module makes a distinct contribution.

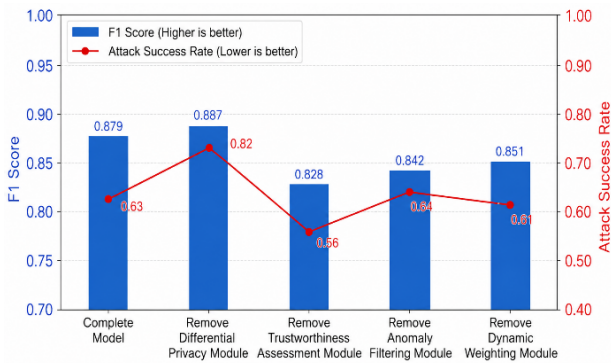


Fig. 8. Model Performance under Different Module-Ablation Settings

4. Conclusions

- (1) This paper addresses data sensitivity, node dispersion, heterogeneity, insufficient trust evaluation, and limited scalability in university sports information systems via a federated learning-based data security and trust evaluation mechanism, combining gradient pruning, DP perturbation, and secure aggregation during local training to reduce leakage risk, plus a four-dimensional trust model incorporated into dynamic aggregation to improve robustness under non-IID and malicious-node conditions. Simulations show the method outperforms Local Training, FedAvg, FedProx, SCAFFOLD, DP-FedAvg, and FLTrust in accuracy, recall, $F1$, and AUC.
- (2) Practical deployment would also need heterogeneous campus IT infrastructure, agreed data schemas, and access-control/audit-logging integration with existing student-information systems—tractable engineering steps. The blockchain, trusted-execution, and personalization directions below are longer-term extensions whose value should be confirmed through real-deployment pilots, not assumed from simulation. These results indicate the method provides a technical solution combining data security, trustworthiness

assessment, and scalable deployment for university sports information systems. Future research will integrate real-world platform data for validation and explore blockchain, trusted execution, and personalized sports recommendations.

- (3) The study remains limited by simulation-based validation. Although the simulated setting follows the structure of university sports information systems, future work should conduct long-term testing with ethically approved, desensitized campus data, and, as a sanity check, evaluate the mechanism on a general-purpose non-IID benchmark such as LEAF [14] to probe generalizability beyond sports-education[cite: 1]. Additional work is needed to study personalized model adaptation, fairness among student groups, and compatibility with blockchain or trusted execution environments under real deployment constraints.

Three further limitations affect how the reported numbers should be read. First, Section 2.12 has no baseline combining differential privacy with multidimensional trustworthiness, so comparisons are against components, not an equivalent integrated system. Second, no physical prototype was built, so end-device memory, server-side computation time, and communication latency under constrained bandwidth are unmeasured; Section 3.5 uses parameter-volume as an overhead proxy instead. Third, trust-weighted aggregation may under-weight merely low-quality nodes (e.g., a small college with an older device) rather than adversarial ones, raising a fairness question this paper does not evaluate. An instrumented prototype, benchmarking against integrated trust+DP baselines, and a fairness audit of weight allocation are the concrete next steps.

References

- [1] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, R. G. L. D'Oliveira, S. El Rouayheb, D. Evans, J. Gardner, Z. Garrett, A. Gascon, B. Ghazi, P. B. Gibbons, M. Gruteser, Z. Harchaoui, C. He, L. He, Z. Huo, B. Hutchinson, J. Hsu, M. Jaggi, T. Javidi, G. Joshi, M. Khodak, J. Konečný, A. Korolova, F. Koushanfar, S. Koyejo, T. Lepoint, Y. Liu, P. Mittal, M. Mohri, R. Nock, A. Ozgur, R. Pagh, H. Qi, D. Ramage, R. Raskar, M. Raykova, D. Song, W. Song, S. U. Stich, Z. Sun, A. T. Suresh, F. Tramer, P. Vepakomma, F. X. Yu, H. Yu, and S. Zhao, (2021) "Advances and Open Problems in Federated Learning"

- Foundations and Trends in Machine Learning** 14(1–2): 1–210. DOI: [10.1561/22000000083](https://doi.org/10.1561/22000000083).
- [2] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. V. Poor, (2021) “Federated Learning for Internet of Things: A Comprehensive Survey” **IEEE Communications Surveys & Tutorials** 23(3): 1622–1658. DOI: [10.1109/COMST.2021.3075439](https://doi.org/10.1109/COMST.2021.3075439).
- [3] V. Mothukuri, R. M. Parizi, S. Pouriyeh, Y. Huang, A. Dehghantanha, and G. Srivastava, (2021) “A Survey on Security and Privacy of Federated Learning” **Future Generation Computer Systems** 115: 619–640. DOI: [10.1016/j.future.2020.10.007](https://doi.org/10.1016/j.future.2020.10.007).
- [4] Z. Lu, H. Pan, Y. Dai, X. Si, and Y. Zhang, (2024) “Federated Learning with Non-IID Data: A Survey” **IEEE Internet of Things Journal** 11(11): 19188–19209. DOI: [10.1109/JIOT.2024.3376548](https://doi.org/10.1109/JIOT.2024.3376548).
- [5] L. Lyu, H. Yu, X. Ma, C. Chen, L. Sun, J. Zhao, Q. Yang, and P. S. Yu, (2024) “Privacy and Robustness in Federated Learning: Attacks and Defenses” **IEEE Transactions on Neural Networks and Learning Systems** 35(7): 8726–8746. DOI: [10.1109/TNNLS.2022.3216981](https://doi.org/10.1109/TNNLS.2022.3216981).
- [6] J. Wen, Z. Zhang, Y. Lan, Z. Cui, J. Cai, and W. Zhang, (2023) “A Survey on Federated Learning: Challenges and Applications” **International Journal of Machine Learning and Cybernetics** 14(2): 513–535. DOI: [10.1007/s13042-022-01647-y](https://doi.org/10.1007/s13042-022-01647-y).
- [7] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer. “Machine Learning with Adversaries: Byzantine Tolerant Gradient Descent”. In: *Advances in Neural Information Processing Systems* 30 (NeurIPS). 2017, 119–129.
- [8] D. Yin, Y. Chen, K. Ramchandran, and P. Bartlett. “Byzantine-Robust Distributed Learning: Towards Optimal Statistical Rates”. In: *Proceedings of the 35th International Conference on Machine Learning (ICML)*. 2018, 5650–5659.
- [9] C. Fung, C. J. M. Yoon, and I. Beschastnikh. “The Limitations of Federated Learning in Sybil Settings”. In: *Proceedings of the 23rd International Symposium on Research in Attacks, Intrusions and Defenses (RAID)*. 2020, 301–316.
- [10] W. Liu, J. Cheng, X. Wang, X. Lu, and J. Yin, (2022) “Hybrid Differential Privacy Based Federated Learning for Internet of Things” **Journal of Systems Architecture** 124: 102418. DOI: [10.1016/j.sysarc.2022.102418](https://doi.org/10.1016/j.sysarc.2022.102418).
- [11] A. P. Kalapaaking, I. Khalil, M. S. Rahman, M. Atiquz-zaman, X. Yi, and M. Almashor, (2023) “Blockchain-Based Federated Learning with Secure Aggregation in Trusted Execution Environment for Internet-of-Things” **IEEE Transactions on Industrial Informatics** 19(2): 1703–1714. DOI: [10.1109/TII.2022.3170348](https://doi.org/10.1109/TII.2022.3170348).
- [12] A. Qammar, A. Karim, H. Ning, and J. Ding, (2023) “Securing Federated Learning with Blockchain: A Systematic Literature Review” **Artificial Intelligence Review** 56(5): 3951–3985. DOI: [10.1007/s10462-022-10271-9](https://doi.org/10.1007/s10462-022-10271-9).
- [13] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang. “Deep Learning with Differential Privacy”. In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 2016, 308–318. DOI: [10.1145/2976749.2978318](https://doi.org/10.1145/2976749.2978318).
- [14] S. Caldas, S. M. K. Duddu, P. Wu, T. Li, J. Konečný, H. B. McMahan, V. Smith, and A. Talwalkar. *LEAF: A Benchmark for Federated Settings*. arXiv:1812.01097 [cs.LG]. 2018. DOI: [10.48550/arXiv.1812.01097](https://doi.org/10.48550/arXiv.1812.01097).
- [15] J. Sun, A. Li, B. Wang, H. Yang, H. Li, and Y. Chen. “Soteria: Provable Defense Against Privacy Leakage in Federated Learning from Representation Perspective”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, 9307–9315. DOI: [10.1109/CVPR46437.2021.00919](https://doi.org/10.1109/CVPR46437.2021.00919).
- [16] N. Bouacida and P. Mohapatra, (2021) “Vulnerabilities in Federated Learning” **IEEE Access** 9: 63229–63249. DOI: [10.1109/ACCESS.2021.3075203](https://doi.org/10.1109/ACCESS.2021.3075203).