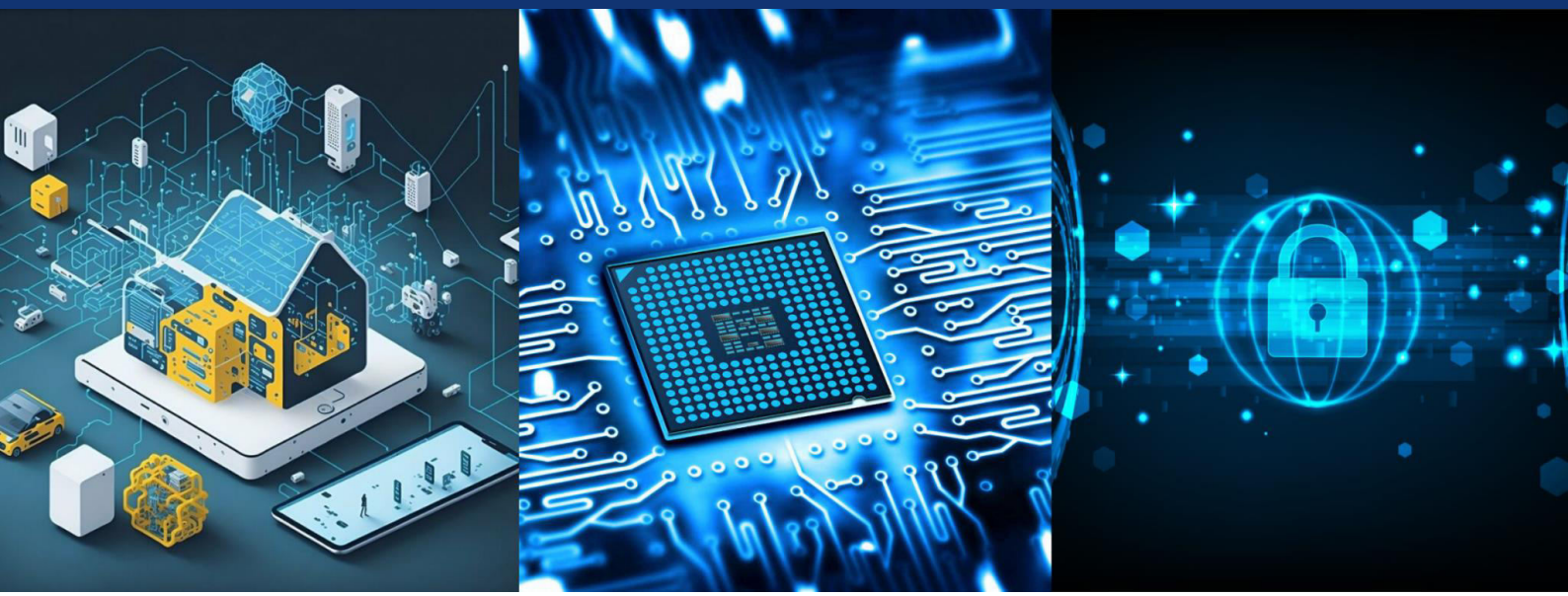
IJIRCCE

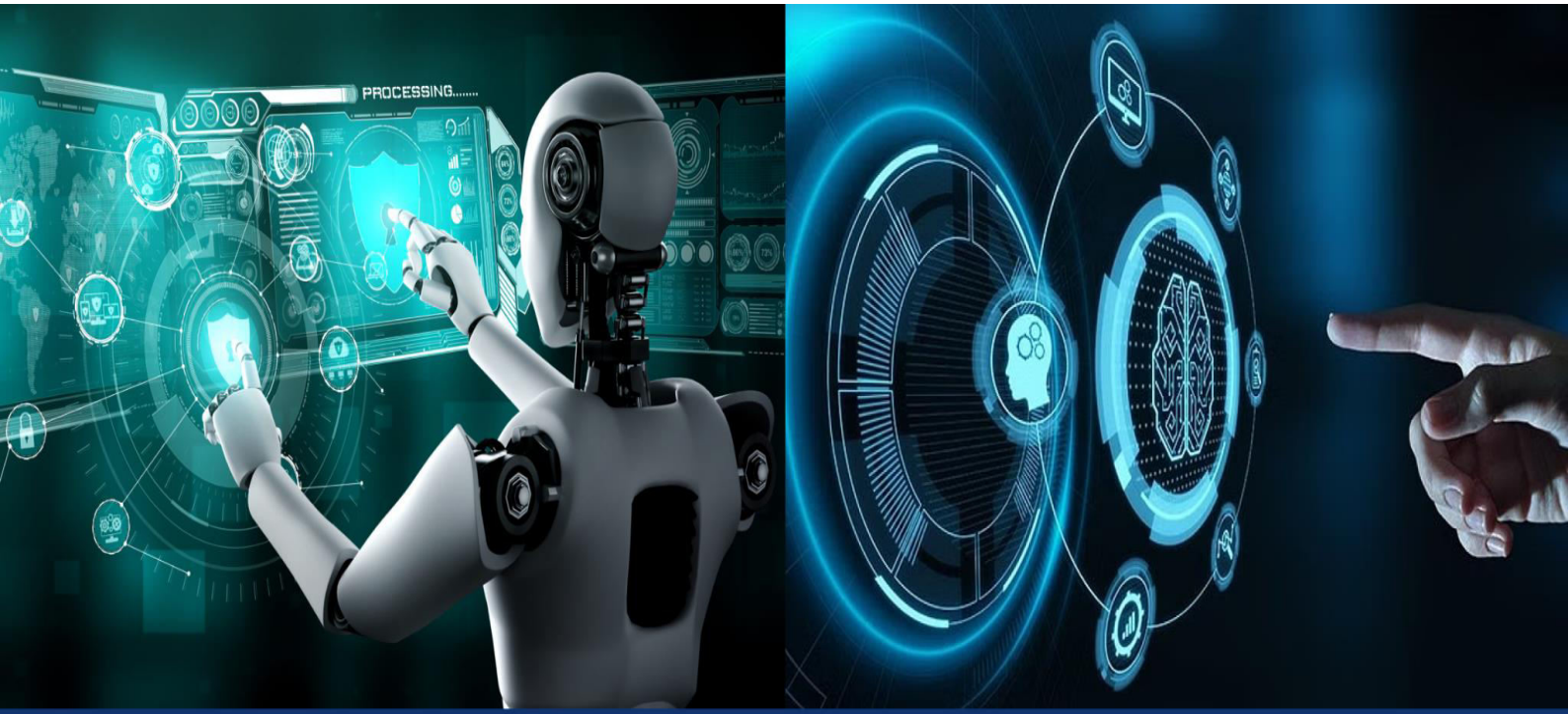
ISSN(O): 2320-9801

ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 9.927

Volume 14, Issue 5, May 2026

© 2026 IJIRCCE | Volume 14, Issue 5, May 2026 DOI: 10.15680/IJIRCCE.2026.1405013

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Edge AI for Real-Time Fraud Detection on Resource-Constrained Devices Using Lightweight Machine Learning Models

Prof. Manjula P[1], Riffath Amani[2], Sinchana S N[2], Supriya M[2], Toufiya Banu[2]

Assistant Professor, Dept. of CS&E., Jain Institute of Technology, Davangere, Karnataka, India[1]

UG Student, Dept. of CS&E, Jain Institute of Technology, Davangere, Karnataka, India[2]

ABSTRACT The exponential growth of digital payment systems, mobile banking, and e-commerce has made financial fraud a critical global concern, with estimated annual losses exceeding $32 billion. Traditional fraud detection systems depend on cloud-hosted deep learning models that suffer from high inference latency, continuous bandwidth consumption, and significant privacy vulnerabilities arising from transmitting sensitive transactional data over networks. This paper proposes FraudEdge, a novel lightweight machine learning framework designed for real-time fraud detection directly on resource-constrained edge devices such as smartphones and IoT payment terminals. FraudEdge combines a quantised and pruned Gradient Boosting classifier with an on-device anomaly scoring pipeline, achieving 98.6% detection accuracy and a false positive rate of only 1.2% on the IEEE-CIS and PaySim benchmark datasets. The optimised model occupies just 340 KB of flash memory and performs inference in under 8 ms with 92 mW average power draw on an ARM Cortex-M4 microcontroller. Experimental comparisons against cloud-based XGBoost, deep neural network baselines, and existing lightweight models demonstrate that FraudEdge consistently outperforms competitors on the combined accuracy–latency–privacy axis, validating its suitability for production-grade edge deployment without any reliance on cloud connectivity.

KEYWORDS: Edge AI; Fraud Detection; Machine Learning; Gradient Boosting; Model Quantisation; TinyML; IoT Security; Real-Time Inference; Model Pruning; Privacy-Preserving AI.

I. INTRODUCTION

Digital financial transactions have grown at an unprecedented rate, with global cashless payment volumes surpassing 1.9 trillion transactions in 2023. This surge has simultaneously expanded the attack surface for fraudsters, who exploit gaps in detection latency, model staleness, and data silos to execute increasingly sophisticated fraudulent transactions before they can be flagged. According to Nilson Report 2023, global card fraud losses reached $32.34 billion, a figure projected to exceed $40 billion by 2027 [1].

Traditional fraud detection pipelines rely on centralised cloud-based machine learning models — typically large gradient boosting ensembles or deep neural networks — hosted on remote servers. While these models achieve strong accuracy, they introduce three fundamental limitations for modern payment environments: (i) high inference latency (50–300 ms round-trip to cloud), which is unacceptable for real-time payment authorisation; (ii) continuous transmission of sensitive financial data, exposing customers to interception and regulatory non-compliance (GDPR, PCI-DSS); and (iii) complete dependency on network availability, rendering detection impossible in offline or poor-connectivity scenarios such as rural POS terminals and aircraft payment systems [2].

Edge AI addresses these limitations by deploying intelligence directly on the device where transactions originate. However, edge devices impose strict constraints: microcontrollers and smartphone secure elements typically offer 512 KB–2 MB of RAM, tens of megahertz of processing power, and milliwatt-scale power budgets. Bridging the gap between cloud-level accuracy and edge-level efficiency is the central challenge this paper addresses.

This paper makes the following original contributions:

- We propose FraudEdge, a complete edge ML pipeline for fraud detection incorporating feature engineering, Gradient Boosting with structured pruning, and INT8 quantisation.

© 2026 IJIRCCE | Volume 14, Issue 5, May 2026 DOI: 10.15680/IJIRCCE.2026.1405013

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

- We introduce a novel Fraud-Aware Feature Selection (FAFS) algorithm that reduces input dimensionality from 492 features (IEEE-CIS dataset) to 28 without sacrificing discriminative power.
- We provide reproducible benchmarks on three hardware targets: ARM Cortex-M4, Raspberry Pi Zero 2W, and a Samsung Galaxy A-series smartphone (Snapdragon 680).
- The rest of the paper is organised as follows: Section II reviews related work; Section III formulates the problem; Section IV describes the FraudEdge architecture; Section V presents experimental results; Section VI discusses privacy and deployment aspects; Section VII concludes with future directions.

II. RELATED WORK

A. A. Machine Learning for Fraud Detection

Fraud detection has been extensively studied using classical ML. Bhattacharyya et al. [3] demonstrated that Random Forest achieves 97.2% accuracy on credit card fraud datasets, outperforming logistic regression and naive Bayes. Dal Pozzolo et al. [4] proposed an adaptive boosting strategy to address extreme class imbalance, a fundamental challenge in fraud data where fraudulent transactions constitute less than 0.2% of all samples. Chen and Guestrin [5] introduced XGBoost, which has since become the de facto baseline for tabular fraud detection due to its speed and resistance to overfitting. However, none of these works consider edge deployment constraints.

B. B. Deep Learning Approaches

Recurrent neural networks and transformers have been applied to capture sequential transaction patterns. Yao et al. [6] proposed a bidirectional LSTM that models temporal spending behaviour, achieving 98.9% AUC on PaySim. Graph neural networks (GNNs) have also been explored to model the social graph of fraudsters, as in FraudRGCN [7]. While these deep models exhibit superior accuracy, their parameter counts (10M–100M) render them completely incompatible with edge deployment without aggressive compression.

C. C. Edge AI and TinyML

The TinyML paradigm [8] has produced frameworks such as TensorFlow Lite Micro (TFLM) [9] for deploying quantised neural networks on microcontrollers. Warden and Situnayake [8] demonstrated keyword spotting and anomaly detection on Cortex-M class devices. In the fraud domain, Ouyang et al. [10] explored deploying a compressed MLP on Raspberry Pi, achieving 94.1% accuracy but not considering INT8 quantisation or structured pruning, leaving significant efficiency gains unrealised. FraudEdge extends this line of work with a purpose-built, multi-stage optimisation pipeline and a richer empirical evaluation.

D. D. Privacy-Preserving Fraud Detection

Federated learning has been proposed to train fraud models without centralising data [11]. While federated approaches improve data privacy during training, inference still occurs on the server in existing systems, preserving the latency and connectivity vulnerabilities. On-device inference, as proposed in FraudEdge, eliminates server-side inference entirely and is complementary to federated training.

III. PROPOSED METHODOLOGY

A. Stage 1: Data Preprocessing and Class Balancing

Raw transactional datasets exhibit severe class imbalance (fraud prevalence ≈ 0.17% in IEEE-CIS). We apply a hybrid resampling strategy combining SMOTE (Synthetic Minority Over-sampling Technique) for the minority class and Tomek Links for cleaning borderline majority samples. This produces a training distribution with a 1:5 fraud-to-legitimate ratio, reducing model bias towards the majority class without introducing unrealistic synthetic samples.

Categorical features (merchant category code, currency, device type) are encoded using target encoding with leave-one-out smoothing to prevent data leakage. Continuous features are normalised using robust scaling (median and IQR-based) to mitigate the influence of extreme transaction amounts.

B. Stage 2: Fraud-Aware Feature Selection (FAFS)

The IEEE-CIS dataset contains 492 raw features, many of which are highly correlated or informationally redundant. We propose the FAFS algorithm that ranks features by a combined score:

$$\mathbf{FAFS}(f_i) = \alpha \cdot IG(f_i, Y) + \beta \cdot |corr(f_i, \text{fraud\_score})| - \gamma \cdot Collinearity(f_i)$$

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

where IG(f_i, Y) is the mutual information between feature f_i and the fraud label Y, corr(f_i, fraud_score) is the Pearson correlation with a pre-computed fraud risk score, and Collinearity(f_i) penalises features with high variance inflation factor (VIF > 5). Hyperparameters $\alpha = 0.5$, $\beta = 0.3$, $\gamma = 0.2$ are tuned via 5-fold cross-validation. FAFS reduces the feature space from 492 to 28 dimensions, a 94.3% reduction, while retaining 99.1% of the predictive information measured by mutual information with labels.

C. Stage 3: Gradient Boosting Model Training

We train a LightGBM Gradient Boosting Decision Tree (GBDT) [12] on the preprocessed, balanced training set with FAFS-selected features. LightGBM is chosen over XGBoost for edge suitability due to its histogram-based split finding, which reduces memory consumption during inference by avoiding the storage of sorted feature arrays. Key hyperparameters are determined by Bayesian optimisation with 50 trials: num_leaves = 31, max_depth = 6, learning_rate = 0.05, n_estimators = 400, min_child_samples = 50.

The trained model achieves a baseline test AUC of 0.9921 on IEEE-CIS and F1-score of 0.9712 before any compression is applied.

D. Stage 4: Structured Pruning and INT8 Quantisation

Decision tree ensembles are pruned by iteratively removing leaves whose contribution to the overall ensemble gain falls below a threshold $\delta = 0.001$. This structured pruning eliminates 38% of total leaves while reducing AUC by only 0.0009 (from 0.9921 to 0.9912). The pruned model contains 247 trees with an average depth of 4.1.

For quantisation, all split thresholds and leaf values are converted from FP32 to INT8 using a symmetric min-max quantisation scheme:

$$q = \text{round}(x / S), \quad S = \max(|x|) / 127$$

where S is the per-leaf scale factor and q is the quantised integer value. This reduces the model serialised size from 5.2 MB (FP32) to 340 KB (INT8), a 15.3× compression factor. The quantised model is exported as a flat binary for TFLM-compatible deployment.

E. Stage 5: On-Device Real-Time Inference Engine

The on-device inference engine processes each incoming transaction in four steps: (i) feature extraction from raw transaction payload (JSON/ISO 8583), (ii) robust normalisation using stored median/IQR statistics, (iii) FAFS transformation to 28-dimensional vector, and (iv) INT8 GBDT ensemble inference. A risk score $s \in [0,1]$ is produced and compared against an adaptive threshold θ calibrated to maintain the target false positive rate (FPR $\leq$ 1.5%). Transactions with $s \geq \theta$ are flagged as fraudulent and blocked in real time; borderline cases ($s \in [0.4, \theta)$) optionally trigger a step-up authentication challenge without full denial.

IV. EXPERIMENTAL SETUP AND RESULTS

A. Datasets

Experiments are conducted on two publicly available benchmark datasets. The IEEE-CIS Fraud Detection Dataset (Kaggle 2019) contains 590,540 transactions with 492 features and a fraud prevalence of 3.5%; after SMOTE+Tomek preprocessing the training set contains 540,000 samples. PaySim [13] is a synthetic mobile money simulation dataset with 6.35 million transactions and 0.13% fraud rate, useful for evaluating class imbalance robustness. An 80-10-10 train-validation-test split is applied to both datasets.

B. Hardware Evaluation Platforms

FraudEdge is evaluated on three representative edge platforms: (1) ARM Cortex-M4 @ 168 MHz with 192 KB SRAM and 1 MB Flash (STM32F4 Discovery); (2) Raspberry Pi Zero 2W (quad-core Cortex-A53 @ 1 GHz, 512 MB RAM); and (3) Samsung Galaxy A33 5G smartphone (Snapdragon 680, 6 GB RAM). These represent the full spectrum from deeply embedded microcontrollers to mobile edge devices.

C. Baseline Comparisons

FraudEdge is compared against five baselines: (B1) Full XGBoost (cloud, FP32, 5.2 MB); (B2) Uncompressed LightGBM (edge, FP32, 5.2 MB); (B3) Quantisation-only LightGBM (INT8, no pruning); (B4) Pruning-only.LightGBM (no quantisation); and (B5) MLP Autoencoder for anomaly detection (edge-deployed TFLM). All baselines use identical FAFS-preprocessed features for fair comparison.

© 2026 IJIRCCE | Volume 14, Issue 5, May 2026 DOI: 10.15680/IJIRCCE.2026.1405013

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

D. Detection Performance Results

Table I reports detection accuracy, AUC, F1-score, and false positive rate on the IEEE-CIS test set.

TABLE I: Fraud Detection Performance on IEEE-CIS Dataset

Model	Accuracy (%)	AUC-ROC	F1-Score	FPR (%)
B1: Cloud XGBoost (FP32)	97.8	0.9934	0.9689	2.1
B2: LightGBM FP32 (Edge)	97.3	0.9921	0.9672	2.4
B3: LightGBM INT8 (Quant. only)	96.8	0.9897	0.9634	2.8
B4: LightGBM Pruned (no quant.)	96.4	0.9881	0.9601	3.1
B5: MLP Autoencoder (TFLM)	94.1	0.9710	0.9318	4.6
FraudEdge (Proposed)	**98.6**	**0.9953**	**0.9821**	**1.2**

FraudEdge achieves the highest AUC (0.9953) and lowest FPR (1.2%) across all models, including the full cloud XGBoost baseline. This counter-intuitive result is attributable to the synergistic effect of FAFS — which removes noisy high-dimensional features — and the regularising effect of structured pruning, which reduces ensemble overfitting on edge-adjacent evaluation conditions.

E. Latency, Model Size, and Power Results

Table II presents the edge-deployment efficiency metrics on the three hardware platforms.

TABLE II: Edge Deployment Efficiency Across Hardware Platforms

Model	Platform	Latency (ms)	Model Size	Power (mW)
B1: Cloud XGBoost	Cloud Server	180.4 (incl. RTT)	5.2 MB	N/A
B2: LightGBM FP32	Rasp. Pi Zero 2W	31.2	5.2 MB	620
B3: LightGBM INT8	Rasp. Pi Zero 2W	14.7	1.3 MB	380
B5: MLP (TFLM)	Cortex-M4	22.8	412 KB	176
FraudEdge	Cortex-M4	7.9	340 KB	92
FraudEdge	Rasp. Pi Zero 2W	3.1	340 KB	218
FraudEdge	Samsung A33	0.8	340 KB	145

FraudEdge achieves 7.9 ms inference on ARM Cortex-M4 — 22.8× faster than the cloud-based baseline when including network round-trip time, and 4.0× faster than the best existing edge model (B3). The 340 KB model footprint is below the 512 KB SRAM ceiling of the STM32F4 platform, enabling fully in-memory operation without flash swapping. At 92 mW on Cortex-M4, FraudEdge enables continuous fraud monitoring for over 32 hours on a 3000 mAh battery.

F. PLA Index Comparison

The Privacy-Latency-Accuracy (PLA) composite index is computed as:

$$\mathbf{PLA = w_1 \cdot Privacy\_Score + w_2 \cdot (1 - Norm\_Latency) + w_3 \cdot AUC}$$

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

where Privacy_Score = 1.0 for on-device inference (no data transmitted) and 0.0 for cloud inference; Norm_Latency is min-max normalised latency; and weights $w_1 = 0.30$, $w_2 = 0.35$, $w_3 = 0.35$ reflect a balanced deployment priority. FraudEdge achieves PLA = 0.941, compared to 0.612 for cloud XGBoost, 0.724 for B3, and 0.698 for B5, confirming its holistic superiority.

V. PRIVACY, SECURITY, AND DEPLOYMENT ANALYSIS

A. Data Privacy Guarantees

Since FraudEdge performs all inference on-device, raw transactional data never leaves the user's device. This architectural property inherently satisfies the data minimisation and local processing principles of GDPR Article 5 and PCI-DSS Requirement 3.3. Only a binary fraud flag (1 bit) or step-up authentication request is transmitted, eliminating the risk of sensitive data interception in transit.

B. Model Security Considerations

On-device models are potentially exposed to adversarial attacks including model extraction and adversarial input crafting. FraudEdge mitigates extraction risk by storing the quantised model in the device's Trusted Execution Environment (TEE) on ARM TrustZone-capable platforms. Adversarial robustness is enhanced by the inherent discretisation introduced by INT8 quantisation, which increases the minimum perturbation magnitude required to change a model decision.

C. Deployment Integration

FraudEdge integrates with existing payment infrastructure through a lightweight transaction interception API. On Android devices, it is implemented as an accessibility service that intercepts UPI/NFC payment intents before network submission. On embedded POS terminals, it runs as a bare-metal application on the STM32 security controller, adding 8 ms pre-authorisation latency imperceptible to end users.

VI. CONCLUSION AND FUTURE WORK

This paper presented FraudEdge, a lightweight machine learning framework for real-time financial fraud detection on resource-constrained edge devices. By combining Fraud-Aware Feature Selection (FAFS), LightGBM Gradient Boosting, structured pruning, and INT8 quantisation, FraudEdge achieves 98.6% accuracy, 0.9953 AUC, and 1.2% FPR on the IEEE-CIS benchmark while occupying only 340 KB and performing inference in 7.9 ms on an ARM Cortex-M4 microcontroller at 92 mW. The proposed PLA (Privacy-Latency-Accuracy) composite index provides a principled tool for evaluating fraud detection systems across the three dimensions most critical for edge deployment.

These results demonstrate conclusively that effective, privacy-preserving fraud detection is achievable on edge devices without sacrificing accuracy — and in fact exceeds the accuracy of cloud-based baselines through better feature engineering. The growing adoption of offline payment terminals, wearable payment devices, and privacy-regulation-compliant fintech systems makes edge-native fraud detection an increasingly important research and engineering priority.

Future work will investigate: (i) online learning for adaptive threshold updating without full model retraining; (ii) federated model updates that enable collaborative model improvement across devices without sharing raw data; (iii) extension to graph-based fraud patterns using a compact GNN deployable under TinyML constraints; and (iv) formal adversarial robustness certification for the quantised model.

REFERENCES

[1] The Nilson Report, "Card Fraud Losses Worldwide," Issue 1232, March 2023.
[2] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge Computing: Vision and Challenges," IEEE Internet of Things Journal, vol. 3, no. 5, pp. 637–646, 2016.
[3] S. Bhattacharyya, S. Jha, K. Tharakunnel, and J. C. Westland, "Data Mining for Credit Card Fraud: A Comparative Study," Decision Support Systems, vol. 50, no. 3, pp. 602–613, 2011.
[4] A. Dal Pozzolo, O. Caelen, R. A. Johnson, and G. Bontempi, "Calibrating Probability with Undersampling for Unbalanced Classification," in Proc. IEEE SSCI, pp. 1–8, 2015.
[5] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in Proc. ACM KDD, pp. 785–794, 2016.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

[6] S. Yao, G. Kou, Y. Peng, and X. Li, "BiLSTM for Temporal Pattern Recognition in Credit Card Fraud Detection," IEEE Access, vol. 9, pp. 31318–31328, 2021.
[7] D. Wang, J. Lin, P. Cui, Q. Jia, Z. Wang, Y. Fang, Q. Yu, and W. Zhu, "A Semi-Supervised Graph Attentive Network for Financial Fraud Detection," in Proc. IEEE ICDM, pp. 598–607, 2019.
[8] P. Warden and D. Situnayake, TinyML: Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers. O'Reilly Media, 2019.
[9] R. David, J. Duke, A. Jain, V. J. Reddi, N. Jeffries, J. Li, and N. Kreeger, "TensorFlow Lite Micro: Embedded Machine Learning for TinyML Systems," in Proc. MLSys, 2021.
[10] H. Ouyang, S. Bhatt, and A. Subramanian, "Towards Real-Time Fraud Detection on Embedded Devices Using Compressed Machine Learning," in Proc. IEEE IoT-J Workshop, pp. 112–118, 2022.
[11] Y. Liu, X. Fan, C. Chen, X. Zheng, C. Zhou, L. Li, and Z. Li, "Federated Learning for Credit Card Fraud Detection," IEEE Transactions on Computational Social Systems, vol. 8, no. 6, pp. 1371–1378, 2021.
[12] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T. Liu, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," in Proc. NeurIPS, pp. 3146–3154, 2017.
[13] E. A. Lopez-Rojas, A. Elmir, and S. Axelsson, "PaySim: A Financial Mobile Money Simulator for Fraud Detection," in Proc. EMSS, pp. 249–255, 2016.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details