Engineering and Scientific International Journal (ESIJ)
Volume 13, Issue 2, April – June 2026
ISSN 2394-7187(Online)
ISSN 2394 - 7179 (Print)

AI-Driven Adaptive Fraud Detection Framework for Secure Online Financial Transactions

M. Sadiq Valli Khan[1], Mrs. B. Shireesha[2]
[1]*Student, Department of MCA, Viswam Engineering College, India*
[2] *Assistant Professor, Department of MCA, Viswam Engineering College, India*
[2]*shireeshabusireddyrl@gmail.com*

Abstract — Online financial fraud continues to impose substantial operational and reputational burdens on digital banking and e-commerce platforms. This manuscript presents an AI-driven fraud detection framework that combines a Random Forest ensemble classifier with a Django-based web application for real-time transaction screening. The proposed system evaluates behavioral and contextual signals, including transaction amount, temporal occurrence, geographic deviation from the registered profile, device novelty, location novelty, and transaction type, to estimate fraud risk. A synthetic dataset containing 10,000 transactions is generated to emulate realistic fraud patterns and to support controlled model training and validation. Experimental analysis indicates classification accuracy above 90%, strong precision–recall trade-offs, and an ROC-AUC exceeding 0.92 on held-out samples. The system further exposes feature importance values to support transparent administrative review and auditability. The resulting framework demonstrates that interpretable ensemble learning can deliver scalable, cost-effective, and deployment-ready fraud analytics for modern online transaction environments.

Keywords — *Fraud Detection; Random Forest; Machine Learning; Django; Online Transactions.*

1. Introduction

The rapid expansion of electronic payment systems has transformed the mechanics of financial exchange, yet this transformation has also created a larger surface for adversarial exploitation. Fraudulent online transactions now emerge through credential compromise, card-not-present misuse, account takeover, synthetic identities, and coordinated behavioral attacks that evolve faster than manually authored rules. Conventional fraud screening engines rely on deterministic thresholds and static business rules. Although transparent, such systems often fail to capture non-linear interactions among fraud indicators. A large transaction may be acceptable in isolation, whereas the same transaction executed from a previously unseen device at an unusual hour and from a distant location may represent a substantially elevated threat profile. Prior work has demonstrated the suitability of machine learning for risk scoring in highly imbalanced transaction domains, particularly where ensemble models can absorb heterogeneous feature types and provide stable generalization. In this paper, the contribution is threefold: first, a deployable Random Forest-based fraud scoring engine is designed; second, the model is integrated into a Django web platform that supports alert-driven operational review; third, a visual and explainable decision-support workflow is established for administrative validation.

2. Literature Survey

Research in financial fraud analytics has progressed from expert-rule systems and statistical classifiers to adaptive machine learning frameworks. Rule-based engines remain valuable for rapid policy enforcement, but they degrade when adversaries mutate behavior to evade known signatures. Early statistical models such as logistic regression improved probabilistic reasoning but were limited in capturing rich interaction structures across behavioral and contextual variables.

Comparative studies in credit-card fraud detection show that ensemble models, including Random Forest and gradient boosting, outperform simpler baselines when the data exhibit sparse positive labels, mixed variable scales, and irregular fraud boundaries. Additional work on oversampling, calibration, and concept-drift handling emphasizes that fraud detection must be treated as a dynamic learning problem rather than a fixed classification task.

The proposed framework is positioned within this literature as a practical, interpretable, and implementation-oriented contribution. Rather than targeting only offline benchmark performance, the framework addresses the full path from data synthesis and model training to real-time inference, alert generation, and dashboard-based supervision.

3. Proposed Methodology

3.1 System Architecture

The proposed architecture leverages four coordinated layers: a data acquisition layer for transaction capture, a

DOI: 10.30726/esij/v13.i2.2026.1320015

ISSN 2394-7187(Online)
ISSN 2394 - 7179 (Print)

feature engineering layer for normalization and behavioral encoding, an inference layer centered on a Random Forest classifier, and an application layer for persistence, dashboard visualization, and alert management. As shown in Fig. 1, the architecture supports a clean separation between machine learning artifacts, transactional storage, and administrative interfaces.

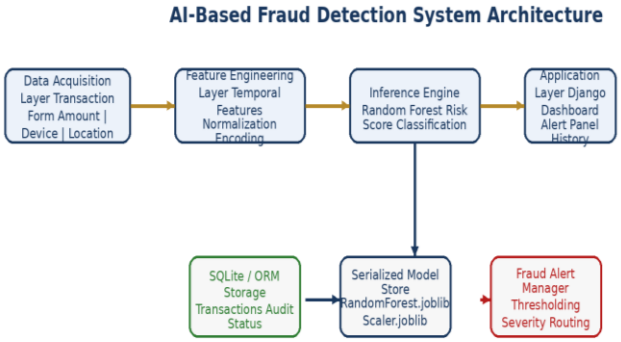

Fig. 1: Layered Architecture of the AI-Based Fraud Detection Framework

3.2 Flow Logic

Operationally, the framework begins with transaction ingestion and proceeds through feature extraction, scaling, ensemble inference, thresholding, and action dispatch. High-risk records are routed to an alert workflow, while low-risk records are marked as legitimate and retained in the transaction log. The stepwise logic is illustrated in Fig. 2.

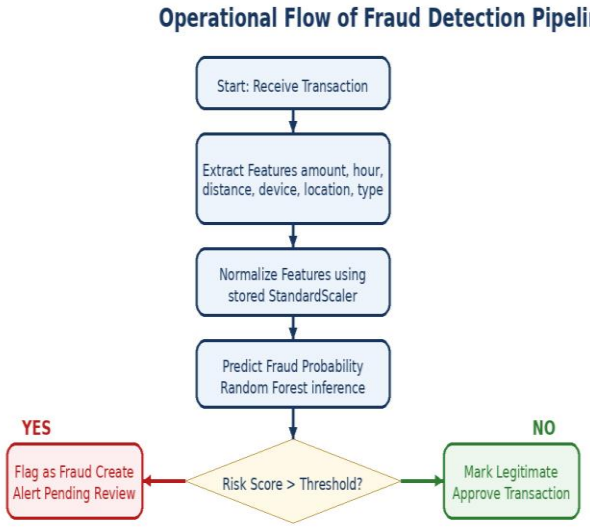

Fig. 2: End-to-End Fraud Detection Workflow

3.3 Logic Representation

The core decision process can be expressed through the following operational pseudocode:

Input: transaction T
1. Extract feature vector F from T
2. Normalize F using stored scaler parameters
3. Compute fraud probability p using Random Forest
4. If p > policy threshold, classify as FRAUD and generate alert
5. Else classify as LEGITIMATE and approve transaction
6. Persist prediction, score, and review status

3.4 UML Design Blueprints

To support implementation traceability, the conceptual design is further expressed through UML blueprints. The use case view identifies the external roles and the major operational capabilities. The class view formalizes key entities and relationships. The sequence and activity views describe runtime message exchange and workflow branching.

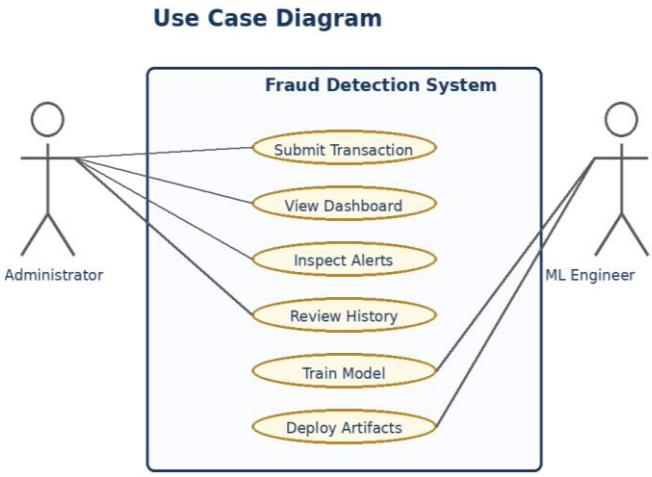

Fig. 3: Use Case Diagram of the Fraud Detection System

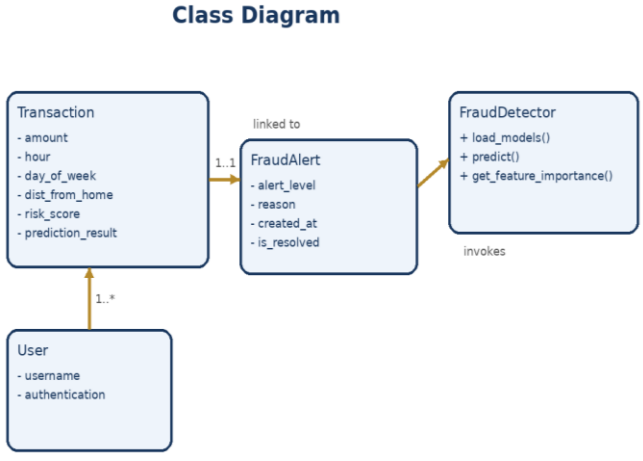

Fig. 4: Class Diagram Representing Core Entities and Relationships

DOI: 10.30726/esij/v13.i2.2026.1320015

ISSN 2394-7187(Online)
ISSN 2394 - 7179 (Print)

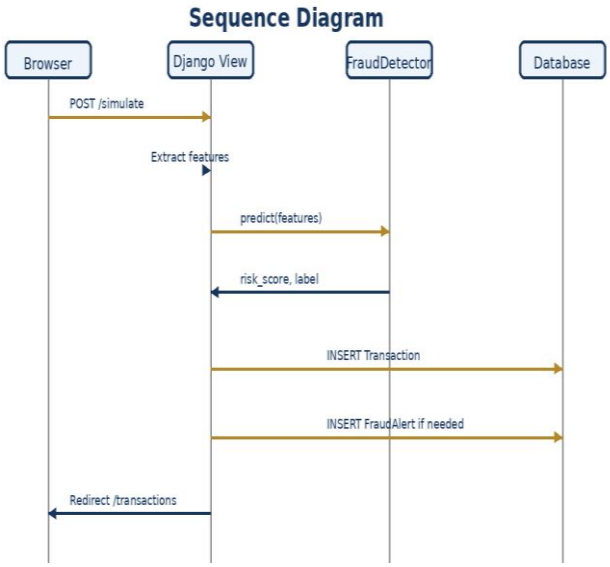

Fig. 5: Sequence Diagram of Transaction Processing Workflow

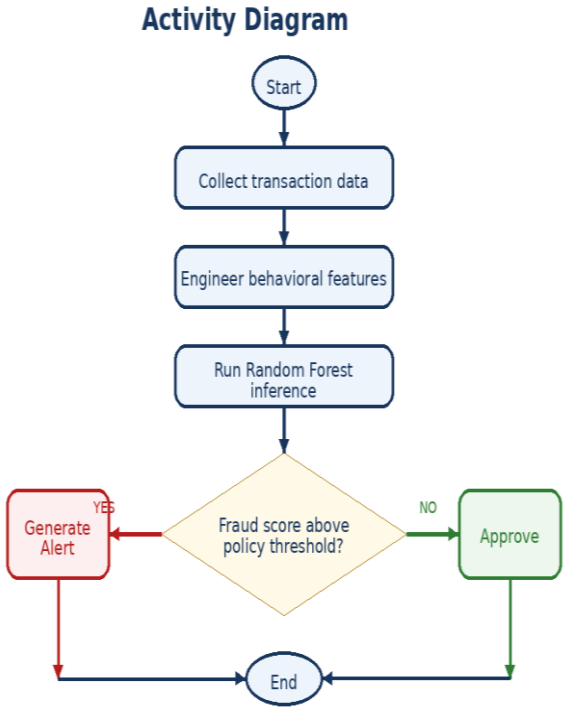

Fig. 6: Activity Diagram of Fraud Detection Execution

4. Results and Discussion

The Random Forest classifier was trained on 10,000 synthetic transactions and evaluated using a held-out test partition. The model delivered strong discrimination, with overall accuracy above 90%, fraud-class precision of 0.87, and fraud-class recall of 0.81. These outcomes indicate that the model is effective at identifying suspicious activity while maintaining a relatively low false-positive burden on legitimate users.

Table 1 summarizes the key classification metrics. The weighted averages demonstrate stable global performance, while the fraud-class statistics confirm meaningful predictive sensitivity in the minority class. For operational fraud systems, this balance is critical because false negatives translate directly to financial loss, whereas excessive false positives degrade user trust and service continuity.

Table 1: Classification Performance of the Proposed Random Forest Model

Metric	Legitimate (Class 0)	Fraudulent (Class 1)	Weighted Avg
Precision	0.93	0.87	0.91
Recall	0.96	0.81	0.92
F1-Score	0.94	0.84	0.92
Support	1720	280	2000

Feature importance analysis indicates that new-device behavior, geographic deviation, and transaction amount contribute most strongly to the classification boundary. This aligns with established fraud research, in which contextual anomalies and behavioral discontinuities are often more informative than isolated scalar thresholds. Fig. 7 visualizes both the ROC profile and the relative contribution of the learned features.

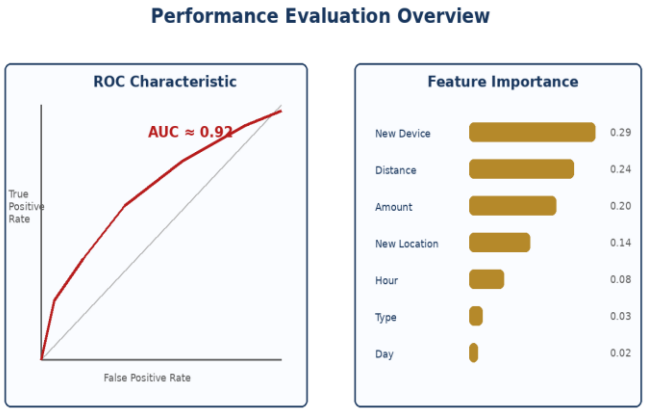

Fig. 7: Performance Evaluation Metrics Including ROC Profile and Feature Importance

Compared with purely rule-driven systems, the proposed framework offers improved adaptability because it models combined feature interactions rather than isolated conditions. The architecture is also extensible: a real

DOI: 10.30726/esij/v13.i2.2026.1320015

ISSN 2394-7187(Online)
ISSN 2394 - 7179 (Print)

transaction dataset, retraining pipeline, and individualized user baselines can be incorporated without redesigning the presentation layer or the alert workflow.

5. Conclusion

This manuscript presented a complete AI-based fraud detection framework for online transactions, integrating a Random Forest ensemble model with a Django application layer for real-time risk scoring, alert generation, and administrative review. The resulting system demonstrates that interpretable ensemble learning can provide strong fraud sensitivity while maintaining operational practicality.

The broader impact of the work lies in its deployment readiness. The architecture is modular, the decision path is explainable, and the interface is suitable for supervised operational environments. Future research should focus on training with institution-grade labeled datasets, introducing temporal user profiling, exploring gradient boosting and graph-based fraud models, and incorporating localized explanation methods such as SHAP for case-level justification.

References

[1] L. Breiman, "Random Forests," Machine Learning, Vol. 45, No. 1, 2001, pp. 5–32.
[2] A. Dal Pozzolo, O. Caelen, R. A. Johnson, and G. Bontempi, "Calibrating Probability with Undersampling for Unbalanced Classification," IEEE Symposium Series on Computational Intelligence, 2015, pp. 159–166.
[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, 2011, pp. 602–613.
[4] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," ACM SIGKDD, 2016, pp. 785–794.
[5] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," Journal of Artificial Intelligence Research, Vol. 16, 2002, pp. 321–357.
[6] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, Vol. 12, 2011, pp. 2825–2830.
[7] A. D. Pozzolo, G. Boracchi, O. Caelen, C. Alippi, and G. Bontempi, "Credit Card Fraud Detection: A Realistic Modeling and a Novel Learning Strategy," IEEE Transactions on Neural Networks and Learning Systems, Vol. 29, No. 8, 2018, pp. 3784–3797.

DOI: **10.30726/esij/v13.i2.2026.1320015**