Inspira-Journal of Commerce, Economics & Computer Science (JCECS)
(A National bi-lingual Quarterly Double Blind Open Access Peer Reviewed Refereed Journal)
ISSN : 2395-7069(Print), Impact Factor: (Cosmos 7.650 & I2OR 5.120), Volume 12, No. 04, October-December, 2026, pp. 01-08

A Hybrid Deep Learning Approach for Real-Time Banking Transaction Fraud Detection: A Survey and Proposed Architecture

Juhili Chandanshive[1], Supriya Choudhary[2], Mandar Bhoite[3], Devesh Ingale[4] & Prof. Shraddha Rokade[5]

[1,2,3,4,5]B.E. Artificial Intelligence and Data Science, Terna Engineering College, Navi Mumbai, Maharashtra, India.

[*]Corresponding Author: shraddharokade@ternaengg.ac.in

Citation: *Chandanshive, J., Choudhary, S., Bhoite, M., Ingale, D. & Rokade, S. (2026). A Hybrid Deep Learning Approach for Real-Time Banking Transaction Fraud Detection: A Survey and Proposed Architecture. Journal of Commerce, Economics & Computer Science, 12(04), 01–08.*

Abstract

Banking systems are increasingly exposed to fraud in real-time digital transactions. Rule-based and single-model detection systems are reported to struggle with evolving fraud patterns, high false-positive rates and severe class imbalance. Hybrid deep learning (DL) approaches, such as convolutional neural networks (CNN) combined with long short-term memory (LSTM) or bidirectional LSTM (BiLSTM) and attention mechanisms, report strong offline performance, but they are seldom designed or evaluated for real-time deployment. This paper surveys machine learning (ML) and DL techniques for transaction fraud detection and analyses their advantages and limitations. From this survey, it identifies a gap between high-performing offline models and the requirements of real-time, scalable and explainable operation. In response, it proposes a conceptual architecture that combines a PyTorch-based CNN-BiLSTM engine, SHapley Additive exPlanations (SHAP) for explainability, and a dual-agent LangChain framework (a SOC Assistant and a User Verification Agent) for risk-based verification and response. The proposed architecture has not yet been experimentally validated. This paper reports no experimental results, and the evaluation plan is described as future work.

Keywords: *Banking Fraud Detection, Class Imbalance, CNN-BiLSTM, Deep Learning, Explainable Artificial Intelligence, LangChain, Real-Time Processing, SHAP.*

Introduction

The rapid expansion of digital banking has brought convenience to users, but it has also exposed financial systems to increasingly sophisticated fraud. As transaction volumes grow, rule-based and single-model fraud detection systems have been described as rigid, because they rely on fixed thresholds and adapt slowly to evolving fraud patterns [6], [7].

A frequently reported limitation of conventional approaches is a high false-positive rate. This disrupts legitimate customers and burdens security operations center (SOC) analysts. In addition, banking transaction datasets are characteristically imbalanced: legitimate transactions greatly outnumber fraudulent ones. This makes training difficult and biases models toward the majority class.

To address these challenges, researchers have increasingly adopted DL. Hybrid models combine CNNs for local feature extraction with recurrent architectures such as LSTM or BiLSTM, and sometimes attention mechanisms, for temporal dependency modelling. They have reported strong offline detection performance [1], [2]. However, the reviewed literature indicates that such models are rarely designed or tested for the low-latency, scalable, real-time deployment that live banking environments

Copyright © 2026 by Author's and Licensed by Inspira. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work properly cited.

require [2], [7]. Furthermore, the growing use of AI in financial decision-making creates a need for interpretable and auditable outputs, which “black-box” DL models often do not provide [6].

This paper makes three contributions: (1) it surveys existing ML, DL and hybrid fraud detection approaches [1]–[8]; (2) it identifies a gap between offline performance and practical real-time, scalable, explainable deployment; and (3) it proposes a conceptual architecture that combines a CNN-BiLSTM model, SHAP explainability and dual-agent verification. The proposed architecture is a design proposal only and has not been experimentally validated.

The remainder of this paper is organised as follows. Section II defines the problem domain. Section III describes the survey methodology. Section IV reviews the literature. Section V compares approaches. Section VI states the research gap. Sections VII and VIII present the proposed architecture and methodology. Sections IX to XI discuss challenges, future work and conclusions.

Background and Problem Domain

To evaluate the surveyed literature, the main technical components and challenges of modern fraud detection pipelines are defined in Table I.

Table I: Key Concepts In Banking Transaction Fraud Detection

Concept	Description
Banking transaction fraud	Unauthorised execution of financial transfers using stolen credentials, compromised devices or manipulated network interactions.
Class imbalance	Fraudulent transactions form a very small fraction of total traffic, which can bias DL models toward the majority class. Common remedies include oversampling, undersampling and GAN/VAE-based synthetic data generation.
CNN feature extraction	Convolutional neural networks (CNNs) extract local patterns and non-linear relationships from dense transaction and behavioural features (e.g., geolocation).
LSTM/BiLSTM temporal modelling	Long short-term memory (LSTM) networks and bidirectional variants (BiLSTM) analyse the chronological sequence of user actions to detect temporal anomalies.
Real-time processing	The requirement to evaluate transactions, produce risk scores and execute decisions within very short time bounds (milliseconds to sub-second) using optimised asynchronous architectures.
Explainable AI (XAI) / SHAP	Methods that decompose model outputs and identify the features that contributed to a given risk score. SHAP is based on Shapley values.

GAN: generative adversarial network. VAE: variational autoencoder. SHAP: SHapley Additive exPlanations.

Survey Methodology

This survey evaluates the landscape of digital banking fraud detection by grouping existing research into three categories: (1) hybrid and temporal DL approaches, (2) ensemble-based approaches, and (3) explainable, survey and review-based approaches.

The works are compared along these dimensions: predictive accuracy in pattern recognition; ability to model temporal data; false-positive handling; resilience to class imbalance; computational cost; and suitability for real-time streaming deployment. Because accuracy alone is insufficient for imbalanced fraud data, Table III also records precision, recall, F1-score, ROC-AUC/PR-AUC and the dataset. Values not available in the source material are marked NR (Not Reported).

Literature Review

The literature shows a progression from traditional ML ensembles to more complex DL hybrids. Significant trade-offs remain between predictive accuracy and practical deployability. Table II summarises the reviewed works, and Table III records the evaluation information they report.

- **Hybrid and Temporal Deep Learning Approaches**

Jainish and Alwin Infant [1] addressed the difficulty of detecting sequential fraud patterns with an attention-integrated BiLSTM. The BiLSTM maps sequential patterns, and the attention layer focuses on the most informative transactional information. The reported accuracy is approximately 94%. The authors

note high computational requirements for training and inference. This limits suitability for low-latency deployment and directly motivates the latency concern in Section VI.

Reddy et al. [2] proposed a CNN-BiLSTM hybrid to combine spatial and temporal feature extraction. The pipeline cleans and normalises the data, extracts features with kernel principal component analysis, and trains the CNN-BiLSTM on those features. The hybrid improved predictive accuracy for sequential transaction patterns. However, its high computational cost limits its applicability in sub-second real-time streaming. This reinforces the gap between offline performance and real-time suitability.

- **Ensemble-Based Fraud Detection**

Khalid et al. [3] combined Random Forest, Gradient Boosting and neural networks to address evolving fraud patterns. They report approximately 93% accuracy and robustness to evolving patterns, but performance is highly data-dependent and sensitive to feature selection.

Bagga et al. [4] built an ensemble learning pipeline that balances fraud detection against false-positive rates. They report approximately 90% accuracy, and the findings depend strongly on dataset-specific characteristics.

Carbo-Valverde et al. [5] used random and causal forests to model non-linear banking customer behaviour. They report approximately 88% accuracy. As with other ensemble methods, generalisability across diverse banking platforms remains a limitation.

Together, [3]–[5] show that ensembles can handle non-linear behaviour and balance false positives. However, their reported dependence on the dataset and on feature selection reduces confidence in how well they generalise.

- **Explainable, Survey and Review-Based Approaches**

Mienye and Jere [6] reviewed DL algorithms for credit card fraud detection. They compare CNN, RNN, LSTM and GRU models, and they discuss suitable performance metrics, training challenges and potential solutions. They highlight open challenges, including temporal dependency modelling. As a review, it does not itself propose a deployable system.

Btoush et al. [7] systematically reviewed ML/DL approaches and identified hybrid and RNN/CNN models as effective for subtle fraud detection. They noted a lack of practical experiments and real-time deployment testing in the literature, which is evidence for the gap in Section VI.

Sharifani and Amini [8] summarised ML and DL methods and emphasised the effectiveness of hybrid models. They acknowledge that their study contains no original experimentation.

Table Ii: Summary Of Reviewed Works on Banking Fraud Detection

Ref.	Approach	Problem Addressed	Reported Finding	Reported Limitation
[1]	Attention-integrated BiLSTM	Sequential/temporal pattern detection	Approx. 94% accuracy	High computational cost
[2]	CNN-BiLSTM hybrid	Joint spatial and temporal features	Improved accuracy on sequential patterns	High training/inference cost; limited sub-second use
[3]	Ensemble (RF, GB, NN)	Evolving fraud patterns	Approx. 93% accuracy; robust to evolving patterns	Data-dependent; sensitive to feature selection
[4]	Ensemble learning pipeline	Fraud detection vs. false positives	Approx. 90% accuracy	Strongly dataset-specific
[5]	Random and causal forests	Non-linear customer behaviour	Approx. 88% accuracy	Limited generalisability across banks
[6]	Review of DL algorithms	DL approaches and challenges	Highlights temporal dependencies and open issues	Review only; no deployable system proposed

[7]	Systematic review (ML/DL)	Effective approaches for subtle fraud	Hybrid and RNN/CNN models effective	No real-time deployment testing
[8]	ML/DL literature review	Conventional vs. advanced models	Summarises ML/DL; highlights hybrids	No original experimentation

RF: Random Forest. GB: Gradient Boosting. NN: neural network.

Table III: Evaluation Information Reported for the Reviewed Works

Ref.	Accuracy	Precision / Recall / F1	ROC-AUC / PR-AUC	Dataset	Imbalance Handling	Computational Cost	Real-time Suitability
[1]	≈94%	NR	NR	NR	NR	High	NR
[2]	NR	NR	NR	NR	NR	High	Limited sub-second suitability
[3]	≈93%	NR	NR	NR	NR	NR	NR
[4]	≈90%	NR	NR	NR	NR	NR	NR
[5]	≈88%	NR	NR	NR	NR	NR	NR
[6]–[8]	Reviews	–	–	–	–	–	[7]: no real-time testing

NR: Not Reported in the uploaded manuscript.

Comparative Analysis

Table IV compares the dominant methodology families using only characteristics stated in the reviewed literature or in the proposal. No numeric or High/Low ratings are used, because the source manuscript provides no criterion for them.

Rule-based systems are generally regarded as fast and scalable, but their fixed thresholds limit temporal and novel-pattern detection. Ensemble ML approaches [3]–[5] address non-linear behaviour and false-positive balance, but they are data-dependent and are not designed to learn deep sequential dependencies. Standalone and hybrid BiLSTM-based models [1], [2] model sequential (and, in [2], combined feature) patterns, but their computational cost limits real-time use. Explainability was not a reported focus in [1]–[5], and black-box behaviour is a recognised concern in DL [6].

The proposed architecture (Section VII) is conceptual. It intends to combine spatial and temporal modelling with low-latency ingestion, explainability and agent-based verification. Whether it can do so is not yet established.

Table IV: Qualitative Comparison of Methodology Families

Approach	Modelling Capability (from Literature)	Reported/known Limitation	Explainability	Real-Time Status
Rule-based	Fixed-threshold rules	Rigid; limited temporal detection	Transparent rules	Fast
Ensemble ML [3]–[5]	Non-linear behaviour; false-positive balance	Data-dependent; limited generalisability	Not a reported focus	NR
BiLSTM + attention [1]	Sequential patterns	High computational cost	Not a reported focus	NR
CNN-BiLSTM [2]	Combined spatial and temporal features	High cost; limited sub-second use	Not a reported focus	Limited
Proposed (conceptual)	Planned CNN-BiLSTM	Unvalidated	Planned (SHAP)	Planned; not evaluated

NR: Not Reported.

Research Gap

The reviewed literature shows that the algorithmic sophistication of fraud detection models has increased, while evidence of their operational deployability has lagged. Hybrid DL approaches such as CNN, LSTM/BiLSTM and attention-based models report strong offline accuracy [1], [2]. However, the

reviewed works report high computational cost [1], [2], and the surveys note a lack of practical, real-time deployment testing [7], [8]. The central gap is therefore the difference between high-performing offline models and practical, scalable, explainable, real-time deployment.

Based on this survey, the reviewed literature appears to lack end-to-end systems that jointly address the following:

- **Low-latency processing:** reducing the computational overhead of DL models so that scores can support real-time transaction decisions [1], [2].
- **Imbalance handling:** moving beyond basic oversampling such as SMOTE toward GAN/VAE-based synthetic data generation in DL pipelines
- **Explainability for compliance:** translating model outputs into feature-level insights for SOC analysts, for example via SHAP [6].
- **Agentic verification:** challenging suspicious transactions through automated verification rather than relying solely on binary thresholds.

This gap motivates the proposed design in Section VII, which aims to address low-latency processing, scalability, class imbalance, explainability and risk-based verification. No claim is made that the design resolves them; that requires the evaluation described in Section X.

Proposed System Architecture

This paper proposes a conceptual, real-time, AI-driven banking fraud detection architecture (Fig. 1). It has four functional layers: Data and Presentation, Data Ingestion and Session, AI Analysis and Verification, and Action and Persistence. The layers are intended to integrate transaction capture, asynchronous processing, hybrid DL inference, explainability, agent-based verification, automated response and secure persistence. The architecture has not been implemented or experimentally validated.

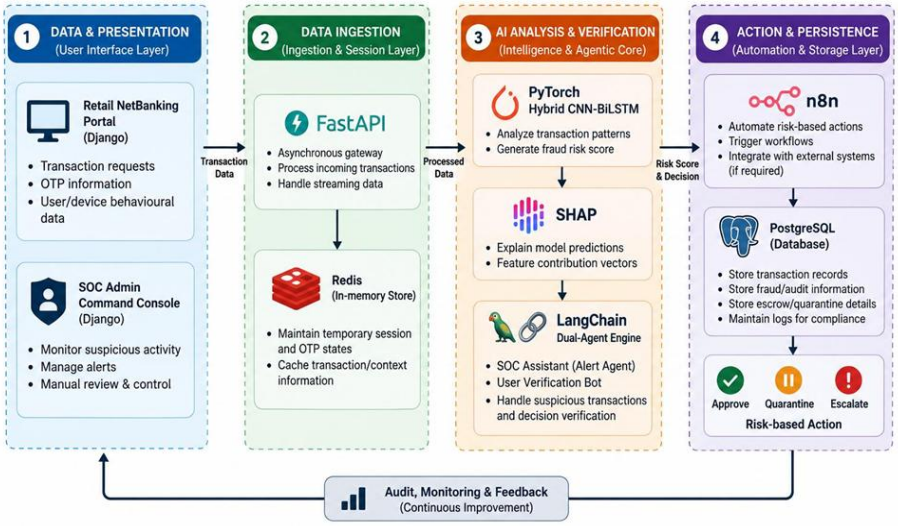


Fig. 1: Proposed system architecture for real-time banking transaction fraud detection

The intended roles of the components in Fig. 1 are as follows.

- **Data and Presentation layer.** A Django-based retail net-banking portal captures transaction requests, OTP information and user/device behavioural data. A Django-based SOC admin command console is intended for monitoring suspicious activity, managing alerts and manual review and control.

- **Data Ingestion and Session layer.** A FastAPI asynchronous gateway receives incoming transactions and handles streaming data. Redis is intended as an in-memory store for temporary session and OTP state and for caching transaction/context information.
- **AI Analysis and Verification layer.** A PyTorch hybrid CNN-BiLSTM model is intended to analyse transaction patterns and generate a fraud risk score. SHAP is intended to explain model predictions through feature-contribution vectors. A LangChain dual-agent engine, comprising a SOC Assistant (alert agent) and a User Verification Agent, is intended to handle suspicious transactions and decision verification.
- **Action and Persistence layer.** n8n is intended to automate risk-based actions, trigger workflows and integrate with external systems where required. PostgreSQL is intended to store transaction records, fraud/audit information, escrow/quarantine details and compliance logs. The risk-based actions are Approve, Quarantine and Escalate.

An audit, monitoring and feedback loop returns information to the presentation layer for continuous improvement. Its mechanism is a design intent and has not been evaluated.

Proposed Methodology

The proposed workflow (Fig. 2) intends to route each transaction according to its assessed risk. Low-risk transactions follow a direct approval path. High-risk transactions undergo explainability analysis and additional verification before a final action is taken.

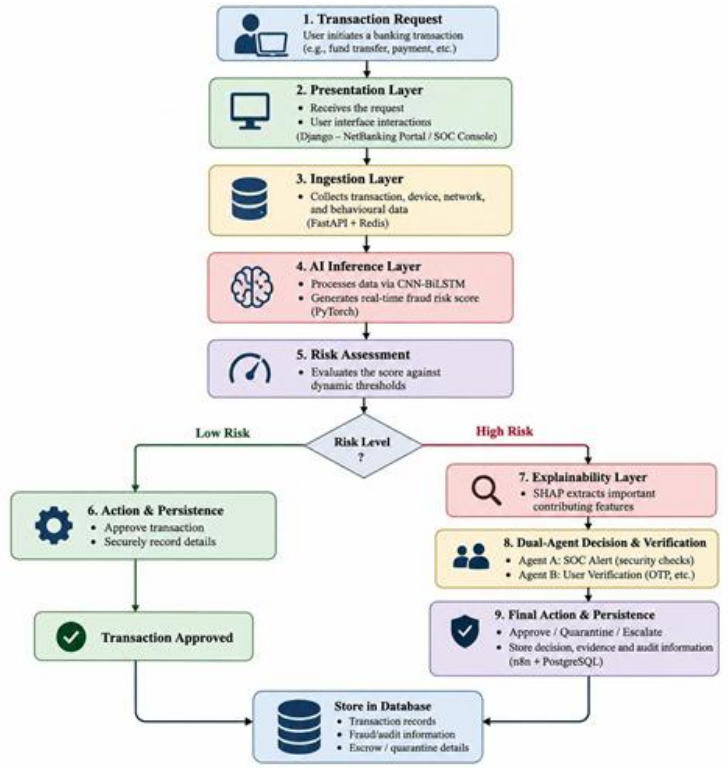


Fig. 2: Proposed methodology for real-time banking transaction fraud detection

The proposed sequence is:

- A user initiates a banking transaction (e.g., fund transfer or payment).
- The presentation layer (Django net-banking portal/SOC console) receives the request.
- The ingestion layer (FastAPI and Redis) collects transaction, device, network and behavioural data.

- The AI inference layer processes the data with CNN-BiLSTM (PyTorch) and generates a real-time fraud risk score.
- The risk assessment step evaluates the score against dynamic thresholds. The method for setting these thresholds is not yet defined.
- **Low risk:** the transaction is approved and recorded securely.
- **High risk:** the explainability layer uses SHAP to extract the most influential contributing features.
- The dual-agent step follows. Agent A (SOC alert) performs security checks, and Agent B (user verification) performs checks such as OTP verification.
- The final action is Approve, Quarantine or Escalate. The decision, evidence and audit information are stored (n8n and PostgreSQL).

Fig. 2 is a conceptual workflow and is not experimental evidence.

Challenges and Limitations

- **Limitations Observed in Existing Research**
  - **Evolving fraud patterns:** static ML models may degrade as fraudsters change tactics [3].
  - **High computational cost:** advanced DL architectures require substantial resources, which complicates real-time inference [1], [2].
  - **Data dependency:** the performance of ensemble models is reported to depend on the dataset and may not generalise across banking environments [3]–[5].
  - **Limited transparency:** DL models are often regarded as black boxes, which constrains regulatory compliance [6].
  - **Limited deployment evidence:** the literature lacks practical real-time testing [7], [8].
- **Potential Challenges of the Proposed Architecture**

  These are anticipated design challenges, not measured results.
  - **Real-time latency:** SHAP explanation and CNN-BiLSTM inference may add overhead that needs optimisation, for example by invoking SHAP only for high-risk transactions as in Fig. 2.
  - **Agent reliability and security:** the LangChain-based SOC Assistant and User Verification Agent must operate securely and without generative hallucination.
  - **Data privacy and security:** capturing extensive behavioural features requires strict data protection and secure persistence.
  - **Class imbalance and model drift:** training with imbalanced data and keeping the model valid as patterns change need to be addressed in implementation.

Future Scope

The proposed architecture has not yet been experimentally validated. The following are planned future experiments, not completed work:

- **Real-time streaming tests:** measuring latency and throughput under intensive streaming transaction loads.
- **Detection performance:** evaluating accuracy, precision, recall, F1-score and ROC-AUC (and PR-AUC) once datasets are integrated.
- **Imbalance handling:** evaluating GAN/VAE-based synthetic generation of minority fraud data.
- **Agent robustness:** evaluating the dual-agent LLM framework to reduce hallucination during critical verification steps.

Conclusion

This paper surveyed ML, DL and hybrid approaches to banking transaction fraud detection. The surveyed literature reports strong offline performance for hybrid DL models, together with high

computational cost, dataset dependency, limited explainability and limited real-time deployment testing. These findings indicate a gap between high-performing offline models and practical, scalable, explainable real-time deployment.

To respond to this gap, the paper proposes a conceptual architecture that integrates a PyTorch CNN-BiLSTM model, SHAP-based explainability, a FastAPI/Redis ingestion layer and LangChain dual-agent verification. The architecture has not been implemented or experimentally validated, and no performance claim is made for it. Future work will evaluate its latency, throughput and detection performance as outlined in Section X.

References

1. G. R. Jainish and P. Alwin Infant, “Attention layer integrated BiLSTM for financial fraud prediction,” *Multimedia Tools and Applications*, vol. 83, no. 34, pp. 80613–80629, 2024.
2. N. Madhusudhana Reddy, K. A. Sharada, D. Pilli, R. N. Paranthaman, K. S. Reddy, and A. Chauhan, “CNN-bidirectional LSTM based approach for financial fraud detection and prevention system,” in *Proc. Int. Conf. Sustainable Computing and Smart Systems (ICSCSS)*, 2023.
3. A. R. Khalid *et al.*, “Enhancing credit card fraud detection: An ensemble machine learning approach,” *Big Data and Cognitive Computing*, vol. 8, no. 1, 2024.
4. Bagga *et al.*, “Ensemble learning pipeline,” 2020.
5. S. Carbo-Valverde *et al.*, “A machine learning approach to the digitalization of bank customers: Evidence from random and causal forests,” 2020.
6. I. D. Mienye and N. Jere, “Deep learning for credit card fraud detection: A review of algorithms, challenges, and solutions,” *IEEE Access*, vol. 12, pp. 96893–96910, 2024, doi: 10.1109/ACCESS.2024.3426955.
7. E. Btoush *et al.*, “A systematic review of literature on credit card cyber fraud detection using machine and deep learning,” *PeerJ Computer Science*, 2023.
8. Sharifani and Amini, “Machine learning and deep learning literature review,” 2023.
9. Agarwal *et al.*, “Hybrid CNN-BiLSTM-attention,” 2021.
10. Alarfaj *et al.*, “State-of-the-art ML/DL algorithms,” 2022.
11. Bandyopadhyay *et al.*, “Ensemble and clustering-based anomaly detection,” 2021.
12. Berhane *et al.*, “Hybrid CNN-SVM,” 2023.
13. Nama, Obaid, and Alrammahi, “Deep learning plus SVM hybrid,” 2023.

❑❑❑