A Comprehensive Review of Financial Fraud Detection Techniques and the Emerging Role of Adaptive Agentic AI

Dr. Chetan Bulla[1], Dr. Shashikanth Gupta[2], Anita Kori[3], Dr. Manas Ranjan Panda[4], Shivanand Kadakal[5], Girish Wali[6]

[1] Post-Doc Research Scholar, Lincoln University College, Petaling Jaya, Malaysia.

Email: bulla.chetan@gmail.com

[2] Lincoln University College, Petaling Jaya, Malaysia; Center for Research Impact & Outcome, Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, Punjab 140401, India.

Email: shashigupta@lincoln.edu.my

[3] Assistant Professor, Department of Artificial Intelligence and Machine Learning (AIML), Basaveshwar Engineering College, Bagalkot, Karnataka, India.

Email: anitakori9488@gmail.com

[4] Independent Researcher; Partner, Banking & Financial Services, Wipro Consulting, USA.

Email: manaspanda01@gmail.com

[5] Independent Researcher; Senior Business Analyst, Citi Bank, USA.

Email: Shivanand.kadakal@gmail.com

[6] Independent Researcher; Solution Architect, Citi Bank, USA.

Email: waligirish@gmail.com

Abstract: The rapid growth of digital banking, online payments, and real-time financial services has significantly increased financial fraud, creating major economic and operational challenges for banks. To mitigate these risks, a wide range of artificial intelligence (AI) techniques—including rule-based systems, machine learning, deep learning, graph-based, and agent-based models—have been widely adopted. However, many existing approaches remain limited by static behavior, weak adaptability to concept drift, high false-positive rates, and limited transparency in decision-making. This review analyzes peer-reviewed studies published between 2015 and 2025, covering traditional and AI-driven fraud detection methods in banking and finance. We examine key research trends, highlighting the shift toward sequential modeling, graph-based reasoning, and explainable AI, while identifying challenges that affect real-world deployment. The paper further discusses Agentic AI as an emerging paradigm that supports autonomous decision-making, continual learning, and coordinated multi-agent reasoning. Finally, open research challenges and future directions are outlined to support the development of robust, adaptive, and trustworthy fraud detection systems..

Keywords: Fraud Detection, Banking Security, Machine Learning, Deep Learning, Graph AI, Explainable AI, Agentic AI, Financial Crime Analytics.

Introduction

The rapid digitization of financial services has fundamentally transformed the way individuals and organizations conduct monetary transactions. Banking systems have evolved from branch-centric operations to highly interconnected digital ecosystems involving mobile applications, real-time payment networks, digital wallets, and cross-border platforms. While this transformation has significantly improved convenience, speed, and financial inclusion, it has also expanded the attack surface for fraudulent activities. As financial systems become more complex and data-driven, fraud detection has emerged as a critical challenge that demands intelligent, adaptive, and trustworthy solutions [1].

Artificial Intelligence (AI) has played a central role in modern fraud detection systems. Over the past two decades, the field has progressed from static rule-based engines to sophisticated machine learning and deep learning models capable of identifying complex patterns in large-scale transactional data. However, despite these advances, existing AI-driven fraud detection systems still struggle with key issues such as concept drift, lack of autonomy, limited explainability, and poor real-time adaptability. These limitations have motivated a shift toward a new paradigm known as Agentic AI, which introduces autonomous, goal-driven, and self-learning agents capable of continuous decision-making and collaboration [2].

The rapid growth of digital banking and payment platforms—such as online banking, mobile applications, card payments, Unified Payments Interface (UPI), and other real-time payment systems—has transformed how financial transactions are carried out. In countries like India, UPI processes billions of transactions every month, enabling instant fund transfers and seamless user experiences. Similar trends are seen globally with card networks and online payment gateways operating continuously across regions and time zones [3]. While these developments have improved convenience and financial inclusion, they have also increased the risk of financial fraud. One major challenge is high transaction velocity. Real-time payments require fraud detection systems to make decisions within milliseconds, leaving very limited time for complex analysis. This creates latency constraints that restrict how advanced models can be used in live systems [4].

At the same time, the large scale and diversity of transactions increase the likelihood of false positives. Even a small error rate can lead to a large number of legitimate transactions being wrongly flagged, which negatively affects customer experience and increases operational costs for banks. Fraudsters take advantage of this environment by using techniques such as phishing, account takeover, identity theft, synthetic identities, and coordinated fraud networks [5]. As fraud patterns change quickly, static or periodically retrained AI models struggle to keep up. This gap between fast-evolving fraud behavior and slow model adaptation leads to reduced detection performance and delayed responses. As a result, fraud detection is no longer just a technical problem but a broader challenge involving scalability, adaptability, explainability, and regulatory compliance.

1.1 Research Motivation

Although modern AI-based fraud detection systems perform better than traditional methods, they still face important limitations in real banking environments. Most existing systems are designed as linear pipelines that ingest data, generate a risk score, and raise alerts. While effective in controlled settings, these systems often struggle when fraud patterns change quickly, data arrives continuously, and decisions must be made in real time. One key challenge is the limited ability of current models to adapt to evolving fraud strategies. Fraudsters continuously adjust their behavior, while many AI models remain static or depend on periodic retraining. In addition, most fraud detection systems focus

on prediction rather than decision-making. A high-risk score alone does not determine the best action, such as whether to block a transaction, request additional verification, or escalate the case for investigation.

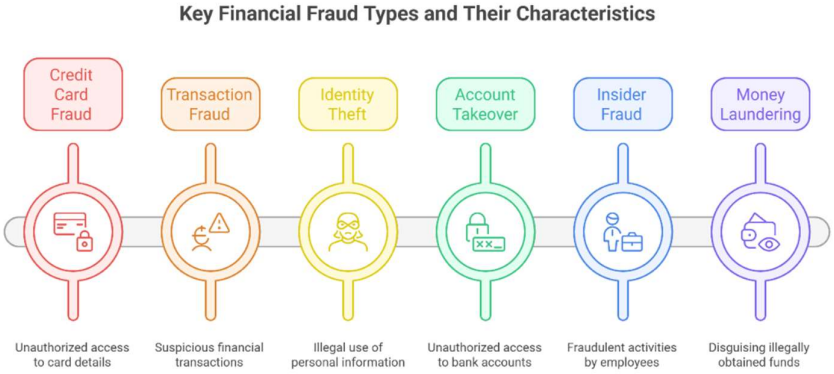

Figure 1: Key financial frauds types and their characteristics

Table 1: Types of Fraud with its complexity and challenges

Fraud Type	**Detection Complexity**	**Key Data Challenges**
Credit Card Fraud	High	Imbalance, real-time constraints
Transaction Fraud	High	Sequential behavior, drift
Identity Theft	Medium–High	Sparse labels, delayed detection
Account Takeover	High	Behavioral mimicry
Insider Fraud	Very High	Limited labels, trust issues
Money Laundering	Very High	Network complexity, regulation

1.2 Objectives and Contributions

The main objective of this review is to provide a structured and up-to-date survey of fraud detection techniques used in banking and finance, and to analyze their evolution from traditional methods to modern AI-driven approaches. The paper reviews rule-based systems, statistical models, machine learning, deep learning, graph-based methods, and explainable AI, highlighting how each approach addresses specific fraud challenges and where they fall short in real-world deployments. Beyond summarizing existing work, this review offers a **comparative perspective** on current fraud detection paradigms. It examines how different generations of techniques handle key requirements such as adaptability to concept drift, real-time decision constraints, explainability, and operational scalability. The central novelty of this survey lies in its treatment of Agentic AI. Unlike prior surveys that present agent-based ideas as isolated modeling techniques, this paper is the first to position Agentic

AI as an architectural evolution in fraud detection. Agentic AI is analyzed as a system-level framework that integrates autonomous decision-making, continuous learning, multi-agent collaboration, and explainability across the entire fraud detection lifecycle.

1.3 From Fraud Types to Agent Capabilities

Across these fraud categories, a key insight emerges: different fraud types require different detection and response capabilities. Single-model approaches are insufficient because they cannot simultaneously handle real-time decisioning, behavioral reasoning, network analysis, and compliance requirements. This diversity motivates an agent-based view of fraud detection, where specialized agents handle detection, investigation, decision-making, learning, and governance. By mapping fraud types to required agent capabilities, this section provides a foundation for understanding why Agentic AI is well suited to modern fraud environments. Instead of treating fraud detection as a single prediction task, it becomes a coordinated system of agents working together to address the full fraud lifecycle.

2. Fraud Detection Techniques

In this section, different fraud detection models are discussed with its categories and limitations. Figure 4 shows the different methods of fraud detection algorithms. Figure show different types of fraud detection techniques that are discussed in this paper : Tradition, Machine Learning and Deep Learning based Fraud detection Techniques

2.1. Traditional Fraud Detection Techniques

Traditional fraud detection in banking and finance was shaped by two practical needs: (i) fast decisioning (often in milliseconds for card authorization and digital payments), and (ii) auditable reasoning that fraud operations teams and compliance groups can understand and defend. Before today's deep learning and graph AI wave, most production anti-fraud stacks relied on rule engines, expert systems, and classical statistical/probabilistic models. Even now, many large banks continue to run hybrid deployments where model scores are combined with expert rules for final decisions (e.g., block, step-up authentication, allow, or queue for manual review)[12].

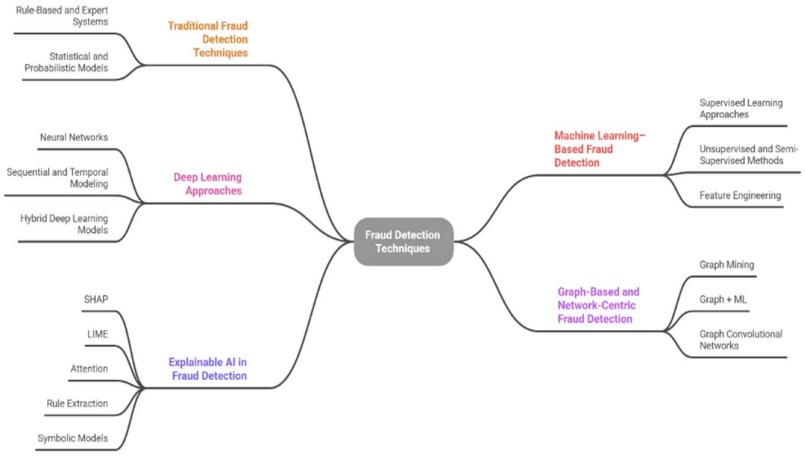

Figure 2: Fraud detection techniques: a comprehensive Overview

Aspect	Rule-Based / Expert Systems	Logistic Regression (LR)	Bayesian Models	Hidden Markov Models (HMMs)
Core Principle	Human-defined rules based on domain expertise and known fraud patterns	Linear probabilistic classification using weighted features	Probabilistic inference using prior knowledge and posterior updates	Sequential probabilistic modeling with hidden behavioral states
Data Requirement	Minimal historical data; relies mainly on expert knowledge	Requires labeled historical transaction data	Can work with limited labels using priors	Requires sequential transaction data per customer/account
Handling of Sequential Behavior	Limited; usually handled via handcrafted velocity rules	Weak; temporal patterns captured only via engineered features	Moderate; supports sequential belief updates	Strong; explicitly models transaction sequences and state transitions
Adaptability to Concept Drift	Low; rules must be manually updated	Moderate; requires periodic retraining	Moderate; posterior updates allow gradual adaptation	Moderate; state transitions can reflect behavior changes but need recalibration
Interpretability	Very high; decisions are fully explainable	High; coefficients indicate feature influence	High; probabilistic reasoning is transparent	Moderate; states are interpretable but transitions may be complex
Computational Complexity	Very low; fast real-time execution	Low; efficient training and scoring	Medium; depends on inference technique	Medium; increases with number of states and features
False Positive Control	Weak at scale; tends to increase over time	Better than rules but sensitive to threshold selection	Improved control through probabilistic risk estimates	Better than static rules due to behavioral modeling
Scalability in Real-Time Systems	High; widely used in production rule engines	High; suitable for high-volume transaction streams	Moderate; may face latency issues in complex models	Moderate; real-time deployment requires optimized implementations

Key Strengths	Immediate deployment, transparency, operational control	Simplicity, robustness, explainability	Explicit uncertainty modeling, adaptive reasoning	Captures temporal and behavioral fraud patterns
Key Limitations	Static, high maintenance effort, high false positives	Limited non-linear modeling capability	Computational overhead, model calibration	Feature engineering complexity, scalability challenges
Representative Applications	Threshold checks, velocity rules, blacklist enforcement	Credit card and transaction fraud scoring	Sequential fraud risk estimation, online monitoring	Account takeover and transaction sequence fraud

Table 2; Comparative analysis of Traditional Fraud detection algorithm

Most importantly, rules alone struggle with **high false-positive rates** in large, high-velocity transaction systems. As transaction volumes grow, even conservative rules can incorrectly block many legitimate transactions, causing customer friction and operational costs. This limitation highlights why traditional techniques, while still necessary, cannot operate effectively on their own and must be complemented by adaptive, learning-based systems.

2.2. Machine Learning–Based Fraud Detection

Machine learning (ML) models remain the most deployed "workhorse" layer in banking and payment fraud detection because they can learn non-linear fraud patterns from transactional, behavioral and contextual features, and can be retrained frequently as fraud tactics evolve.

Supervised Learning Approaches

Supervised learning assumes that historical transactions are labeled (fraud / non-fraud). In finance, this is powerful but challenging due to extreme class imbalance (fraud is rare), delayed labels, and distribution shift (fraud strategies change). Therefore, recent papers emphasize metrics beyond accuracy—precision, recall, F1, ROC-AUC and PR-AUC—and careful sampling / cost-sensitive training [29]

Decision Trees (DT): Decision Trees are popular as interpretable baselines: they produce human-readable rules ("if amount high and merchant risk high, then fraud"). However, single trees often overfit and become unstable under imbalance. Because of this, many recent works use DT mainly for baseline comparison inside a benchmarking study, and as the base learner inside ensembles (Random Forest, Gradient Boosting, XGBoost). The DT usually offers *good interpretability but lower recall* than ensembles on imbalanced credit-card datasets (e.g., Kaggle/ULB), which motivates moving to RF/XGBoost as the "production-grade" choice [29]

Random Forest (RF): Random Forest (RF) is a bagging ensemble of decision trees and is widely used in fraud detection because it handles mixed features and non-linear interactions, relatively robust to noise, and it provides usable **feature importance** for analyst guidance. GA + ML engine (Journal of Big Data, Springer, 2022): Ileberi et al. proposed using a Genetic Algorithm (GA) for feature selection and then trained multiple supervised classifiers (DT, RF, LR, ANN, NB). The study highlights that

feature choice strongly affects fraud detection performance, and positions RF as a strong learner within a feature-optimized pipeline [22]. FraudX AI (Computers, MDPI, 2025): Baisholan et al. proposed an ensemble framework that includes RF and XGBoost, preserves natural imbalance, and adds interpretability using SHAP. They report Recall ≈ 95% and PR-AUC ≈ 97% on the European card dataset, showing that well-tuned ensemble learners can achieve strong rare-class capture with explainability [23]. Interpretable RF vs XGBoost (Applied Sciences, MDPI, 2025): Iqbal et al. compared RF and XGBoost and additionally applied SHAP and LIME. They report RF achieving higher precision (~89.09%) and F1 (~0.9159), while XGBoost achieved higher recall (~95.56%) and ROC-AUC (~0.9997)—a useful observation for banks choosing between "fewer false alerts" vs "catch more fraud."

Support Vector Machines (SVM): SVM remains relevant for fraud detection, especially when feature spaces are well-scaled and the boundary between fraud and non-fraud can be separated with a margin. However, in modern financial datasets with millions of transactions and complex non-linear patterns, SVM can face scalability limits (training cost), sensitivity to kernel/parameter selection, and difficulty handling extreme imbalance unless combined with sampling or cost-sensitive settings. In recent practice-oriented studies, SVM is frequently treated as a benchmark classifier rather than the final best model, with RF/XGBoost usually winning on imbalanced transaction datasets [23]

XGBoost: XGBoost is one of the most consistent "top performers" in tabular fraud detection because boosting focuses on hard-to-classify examples, which often includes fraud cases. It also supports practical controls (regularization, tree depth, learning rate) and integrates well with resampling/threshold tuning. Mobile payment fraud at scale (Information Systems Frontiers, Springer, 2023): Hajek et al. proposed an XGBoost-based fraud detection framework validated on >6 million mobile transactions. Importantly, they evaluate not only standard metrics but also financial cost savings, and show that combining unsupervised outlier detectors + XGBoost can yield the best classification results, while random under-sampling + XGBoost can maximize cost savings. This supports a key real-world point: "best AUC" is not always "best business outcome [24] Hyperparameter-tuned XGBoost [25] studied optimized/hyperparameter-tuned XGBoost for payment fraud prediction, emphasizing how tuning and validation strategies significantly impact outcomes on real transaction-style datasets. As noted earlier, Iqbal et al. report XGBoost producing higher recall and ROC-AUC than RF, making it attractive for fraud teams focused on catching more fraud, while RF yields better precision/F1. XGBoost is often preferred when the institution prioritizes high recall (fraud capture) and has a mature alert management process to handle increased investigation volume; and it becomes even more practical when paired with cost-based thresholds and XAI tools (e.g., SHAP) for auditability.

Model	**Recall (Fraud Capture)**	**Precision / F1**	**ROC-AUC**	**Interpretability**	**Cost Sensitivity**	**Typical Use Case**
Decision Tree (DT)	Low–Medium	Medium	Medium	**High**	Low	Rule-based, regulatory-friendly baselines
Support Vector	Medium	Medium	Medium	Low	Low	Baseline compariso

Machine (SVM)						n with tuned kernels
Random Forest (RF)	Medium	**High**	High	Medium–High (FI, SHAP)	Medium	Precision-driven systems with controlled alerts
XGBoost	**High**	Medium	**Very High**	Medium (with SHAP)	**High**	Cost-sensitive, high-recall fraud detection

Table 3: Comparison of Machine learning based fraud detection algorithms

2.3. Deep Learning Approaches

Deep learning (DL) is now widely used in financial fraud detection because it can learn complex, non-linear patterns directly from raw transaction features, and it can also model behavior changes over time (sequence/temporal learning). Recent Q1–Q3 journal works show three major directions: (i) feed-forward networks for tabular data (DNN/MLP), (ii) CNN/RNN models for local + temporal patterns, and (iii) hybrid/attention models that combine representation learning + sequence focus [41]

Neural Networks for Fraud Detection

DNN (Deep Neural Networks / MLP for tabular transactions): Most banking fraud datasets are tabular (amount, time gaps, merchant category, device/IP, geo, velocity features, etc.). A standard DNN (often called MLP) takes engineered or normalized features and learns decision boundaries without manual rule creation. The DNN is usually trained with class-weighted loss / focal loss for imbalance, SMOTE or other resampling and threshold tuning based on precision–recall trade-off (fraud is rare). **It** Works well as a strong baseline on tabular fraud features and fast inference, easy deployment in scoring pipelines. But without sequence context, it may miss *behavioral* fraud patterns (e.g., "normal-looking" transactions that are suspicious only when seen as a sequence) and it is limited interpretability unless combined with XAI.

Wu et al. (2025) [40] propose a "brain-like" continuous-coupled neural network (CCNN) and compare it with classical ML baselines (DT, KNN, LightGBM, LR, NB). They report near-perfect performance on the Kaggle credit card dataset (after SMOTE and feature-to-matrix transformation) [40]. While CCNN is not a plain MLP, the paper is useful because it shows that deep non-linear learners can exceed classical models, especially when imbalance handling and representation are strong.

CNN (Convolutional Neural Networks for local feature interactions): CNNs in fraud detection are usually 1D CNNs. They treat a transaction's feature vector (or a short window of transactions) as a "signal" and learn local interactions among neighboring features or within a window. CNNs are popular when you transform features into a matrix/grid form (feature maps), or you want a compact encoder for "local" dependencies. Convolutional Neural Networks (CNNs) are efficient feature

extractors due to parameter sharing and local receptive fields, which significantly reduce model complexity while maintaining expressive power. They are particularly effective at learning local feature patterns and short-range correlations in structured or tabular data, making them suitable for capturing spatial or neighborhood-level irregularities without requiring heavy sequential modeling.

Despite their strong local feature learning capability, CNNs are limited in modeling long-term temporal dependencies and sequential relationships inherent in transaction streams. As a result, CNN-only architectures may fail to capture evolving fraud behaviors over extended time horizons, and are often complemented with recurrent networks (LSTM/GRU) or attention-based mechanisms to achieve more comprehensive temporal modeling. In UAAD-FDNet (2023), the authors compare several deep learning baselines (LSTM, CNN, MLP, AE) on Kaggle and IEEE-CIS fraud datasets and show DL approaches generally outperform standard ML baselines (SVM/DT/XGBoost/KNN/RF) in their experiments. MDPI

RNN / LSTM / GRU (sequence-aware transaction modeling): RNN-family models (LSTM/GRU) are applied when fraud detection is framed as: "Given a sequence of recent transactions, predict whether the next transaction (or the current one) is fraudulent." This is more realistic for banking, because fraud often appears as behavior drift (unusual timing, unusual merchant chain, unusual geo sequence).

The performance of LSTM and GRU models is highly sensitive to the definition of the input sequence, including window length, padding strategy, and handling of irregular time gaps between events. Training these models is computationally more expensive than feed-forward architectures such as MLPs or CNNs. Moreover, despite their improved memory mechanisms, they may still struggle to capture very long-range dependencies when critical events occur far back in the sequence, a limitation that is often mitigated by incorporating attention mechanisms. Benchaji et al. (2021) propose an **LSTM + attention** fraud detection system with additional steps like feature selection/dimension reduction and imbalance handling. Their core motivation is that classic models using single transactions ignore the *behavior profile* of a cardholder, while LSTM helps encode sequential dependency.

Sequential and Temporal Modeling

Transaction sequence modeling: Sequence modeling means you do fraud scoring using context: recent amounts, merchants, timestamps, location shifts, device changes, and interaction patterns. Common design choices Sliding windows: last k transactions (e.g., 10, 20, 50), Time-based windows: last t hours/days, and Irregular time gaps: encode inter-event time as a feature (important in real payments) Benchaji et al. (2021) explicitly treat fraud detection as a sequential modeling problem and use attention to focus on the most important transactions inside a sequence [37]. UAAD-FDNet (2023) reframes fraud detection as anomaly detection (fraud = abnormal) and uses an autoencoder-based framework with attention + GAN to better learn the normal transaction distribution under strong imbalance. This is a different but useful temporal idea: instead of directly classifying fraud, you learn what "normal" looks like and flag deviations [38]

Customer-centric learning: Customer-centric learning means the model learns patterns at the cardholder/account level, not only at global transaction level. Typical approaches **Per-customer sequences** (each customer has their own chronological history), **Behavior embeddings** (vector representation of a customer) and **Personal baselines** ("this is unusual for this customer"). This helps especially for high-value customers with stable spending habits, account takeover (sudden behavior shift) and geo/device anomalies. Both the attention-LSTM direction (customer transaction history) and anomaly-learning direction (normal profile reconstruction) naturally support customer-centric fraud detection [38][39]

Hybrid Deep Learning Models

CNN–LSTM: CNN–LSTM hybrids models where CNN extracts compact representations from features or short windows and LSTM/GRU models how those representations evolve across time. CNN reduces noise and learns strong local patterns; LSTM captures behavioral dynamics.

Attention-based models: Attention is used to answer: "Which past transactions in this sequence matter most for deciding fraud now?" . Benchaji et al. (2021) use attention on top of LSTM so the model can weight more important transactions in the sequence, rather than treating every step equally. [37] UAAD-FDNet (2023) uses a feature attention idea inside an unsupervised anomaly framework and reports measurable improvements versus an AE baseline and also compares performance against common DL baselines. In their Kaggle experiments, they state that AE improves over LSTM/CNN/MLP, and UAAD-FDNet improves further over AE; adding feature attention improves further again [38]

Model	Inputs	Focus	Strengths	Limitations	Use Case	Key References
DNN / MLP	Tabular features	Non-linear classification	Fast, strong baseline	No sequence context	Real-time scoring	Wu *et al.*, 2025
CNN (1D)	Feature vectors / windows	Local patterns	Efficient feature extraction	Weak long-term modeling	Hybrid encoders	UAAD-FDNet, 2023
LSTM / GRU	Transaction sequences	Temporal dependencies	Captures behavior drift	Window-sensitive, costly	Sequential fraud	Benchaji *et al.*, 2021
Attention + LSTM	Sequences + attention	Relevant event focus	Better long-range recall	Higher complexity	Burst fraud	Benchaji *et al.*, 2021
Autoencoder (AE)	High-dimensional data	Normality learning	Label-scarce friendly	Threshold tuning	Anomaly detection	Du *et al.*, 2023
AE + Tree	AE embeddings + features	Hybrid learning	Strong recall, robust	System complexity	Semi-supervised pipelines	Shanaa *et al.*, 2025

CNN–LSTM	Windows → sequences	Local + temporal	Noise reduction	Higher compute	End-to-end DL	Benchaji *et al.*, 2021
Customer-Centric	Per-user histories	Personal baselines	ATO / drift detection	Needs history	High-value users	UAAD-FDNet, 2023
Temporal Anomaly	Sequences	Deviation detection	Handles imbalance	No explicit probability	Novel fraud	UAAD-FDNet, 2023

Table 4: Comparison of Deep learning models

Deep learning models have strengthened fraud detection by learning complex patterns directly from data, but their benefits vary depending on the architecture and deployment context. Instead of describing each model separately, this section should emphasize how different deep learning approaches trade accuracy for operational cost.For example, sequential models such as LSTMs and attention-based networks improve recall by capturing transaction order and long-term behavior changes. However, these gains often come at the cost of higher latency, increased computational requirements, and reduced explainability, which can limit their use in real-time fraud systems.

2.4. Graph-Based and Network-Centric Fraud Detection

Graph-based fraud detection approaches model financial systems as interconnected networks rather than as collections of independent transactions. This perspective is essential because many real-world fraud schemes are relational in nature. Examples include mule-account networks, coordinated merchant fraud, phishing clusters, synthetic identity creation, and multi-step money laundering. In such cases, individual transactions may appear legitimate, while fraudulent behavior only becomes evident when relationships among entities are analyzed collectively [43][44]

Graph Representation of Financial Systems: The effectiveness of graph-based fraud detection strongly depends on how the graph is constructed. This includes defining node types, edge types, and feature placement. A commonly used representation is the account–account transaction graph, where nodes represent accounts or cards and edges represent transfers or payments. This structure is effective for identifying money-flow fraud, mule networks, and laundering chains, but it often lacks contextual detail if additional entities such as devices, merchants, or locations are not modeled [44-50]

Graph Mining and Community Analysis

Before the widespread adoption of deep graph learning, fraud detection relied heavily on classical graph mining techniques combined with traditional machine learning models. These methods remain important today, particularly for interpretability and investigative analysis.Community detection is commonly used to identify fraud rings, which often manifest as dense subgraphs formed through shared merchants, devices, or beneficiaries. Modularity-based algorithms such as Louvain clustering are widely applied to uncover such coordinated groups for analyst review [43]. Centrality analysis further helps identify mule accounts or key intermediaries, which often exhibit high betweenness or hub-like behavior. Motif and subgraph pattern mining is also used to detect recurring structures, such as fan-in and fan-out patterns, that are indicative of laundering activity [46][47].

A typical pipeline extracts structural features from graphs, such as degree statistics, clustering coefficients, community identifiers, or random-walk embeddings, and feeds them into classifiers like Random Forest or XGBoost. While effective in some scenarios, this approach often struggles when fraud is extremely imbalanced or when detection depends on multi-hop relationships. These limitations motivate the use of deep graph models that directly learn from neighborhood structure [43][44][47].

Graph Neural Networks

Graph Neural Networks learn node representations through message passing, where each node aggregates information from its neighbors and, in deeper layers, from more distant nodes. This mechanism is well suited for fraud detection because fraud evidence is often distributed across a network rather than localized to a single transaction [43].

In real financial systems, interactions are rarely homogeneous, which motivates the use of heterogeneous GNNs. These models explicitly handle multiple node and edge types using relation-specific transformations or attention mechanisms. Architectures such as Relational GCNs and heterogeneous attention networks have demonstrated superior balance between precision and recall by leveraging relation-aware information. Evidence from enterprise, supply-chain, and blockchain fraud detection shows that heterogeneous GNNs outperform homogeneous graph models and classical baselines [50].

Category	Method / Model	Key Idea	Main Strength	Main Limitation
Concept [43], [44]	Graph-based fraud detection	Models transactions as networks of entities	Captures coordinated & multi-hop fraud	Higher complexity
Graph Design [44]	Account–Account graph	Accounts/cards as nodes; payments as edges	Money-flow & mule detection	Limited context
[45], [50]	Heterogeneous graph	Users, devices, merchants, IPs as nodes	Realistic system modeling	Complex construction
[47]	Transaction-node graph	Transactions as nodes with temporal links	Semi-supervised risk propagation	Large graphs
Features [46]	Node + edge features	Behavioral + transactional attributes	Rich context	Engineering effort
Graph Mining [43]	Communities / centrality	Finds fraud rings & mule roles	Interpretable	Limited learning
Graph + ML [43], [44]	Structural features + RF/XGB	Graph stats → classifier	Simple pipeline	Weak for multi-hop fraud
GNNs [50]	GCN	Uniform neighbor aggregation	Stable baseline	Low recall

[50]	GAT	Attention-weighted neighbors	Better recall / PR-AUC	Noisy attention
[50]	Hetero-GNN (R-GCN, HAN)	Relation-aware learning	Best precision–recall balance	Training cost
Hybrid [49]	GNN + AE + Ensemble	Feature anomaly + network learning	High F1 / AUC	Deployment complexity

Table 5 : Comparison of Graph based fraud detection models

Graph-based fraud detection methods are powerful for identifying organized fraud, but their deployment in real banking systems introduces important operational challenges. One key issue is **graph freshness**. Fraud graphs evolve continuously as new transactions, devices, and accounts appear. Keeping the graph up to date in near real time is difficult, and stale graphs can lead to missed fraud patterns or delayed detection. Another challenge is the **cost of real-time inference**. Graph neural networks and subgraph analysis are computationally expensive, especially at scale. This makes it hard to apply full graph reasoning within the strict latency limits required for real-time payment decisions. As a result, graph models are often used in batch or near-real-time settings rather than in live transaction flows.

Label delay further limits the effectiveness of graph-based learning. In many cases, fraud labels arrive weeks or months after the activity occurs, slowing model updates and making it harder to adapt to new fraud structures. This delay is especially problematic for graph models that depend on accurate labels to learn relational patterns. These constraints show that while graph-based approaches are essential for network-level fraud detection, they must be carefully integrated with other methods and supported by adaptive system designs to be effective in production environments.

2.4. Explainable AI (XAI) in Fraud Detection

Explainable Artificial Intelligence (XAI) has become a fundamental requirement in financial fraud detection systems. Beyond predictive accuracy, financial institutions must ensure that automated decisions are transparent, auditable, and aligned with regulatory and operational constraints. Regulations such as GDPR and supervisory guidelines such as SR 11-7 require that models affecting customers be explainable, well documented, and continuously monitored. Consequently, highly accurate black-box models cannot be deployed in isolation without accompanying explanation and governance mechanisms.

Model-agnostic explainability methods are widely adopted because they can be applied to any predictive model. SHAP has emerged as the most commonly used technique in fraud detection due to its theoretically grounded feature attributions and support for both global and local explanations. Empirical studies show that SHAP can be effectively combined with gradient boosting models, deep networks, and even federated learning frameworks without sacrificing performance [50]. LIME complements SHAP by providing simpler, case-level explanations that are easier for analysts to interpret quickly, although its attributions are less stable across samples [46], [48]. In practice, many systems use SHAP for global understanding and LIME for local decision support.

Overall, current evidence suggests that effective fraud detection systems increasingly integrate multiple XAI techniques. Combining high-performing models with complementary explanation methods allows institutions to balance accuracy, interpretability, and compliance, making XAI a core component of production-grade fraud detection pipelines [50]

XAI Method	Model Type	Explanation Scope	Key Advantage	Key Limitation	Typical Use
SHAP	Model-agnostic	Global + Local	Consistent feature attribution; regulator-friendly	Computational cost	Model validation, audits
LIME	Model-agnostic	Local	Simple, intuitive explanations	Attribution instability	Case-by-case analysis
Attention	Deep learning	Local / Temporal	Highlights important time steps or features	Not fully causal	Sequential fraud detection
Rule Extraction	Hybrid / Symbolic	Global	Human-readable logic	Limited scalability	Regulatory compliance
Symbolic Models	Inherently interpretable	Global	Explicit decision functions	Lower flexibility	High-regulation environments

Table 6: Comparison of Explainable AI methods

3. Limitations of Current AI-Driven Fraud Systems (Analytical Insights)

Even though modern ML/DL (including GNNs) improves detection accuracy, most production fraud stacks in banking still behave like "smart classifiers" rather than autonomous, adaptive decision systems. Recent surveys and reviews repeatedly highlight gaps around drift, real-time constraints, explainability, and operational deployment friction [47]. Figure show limitation of AI Driven Fraud detection models

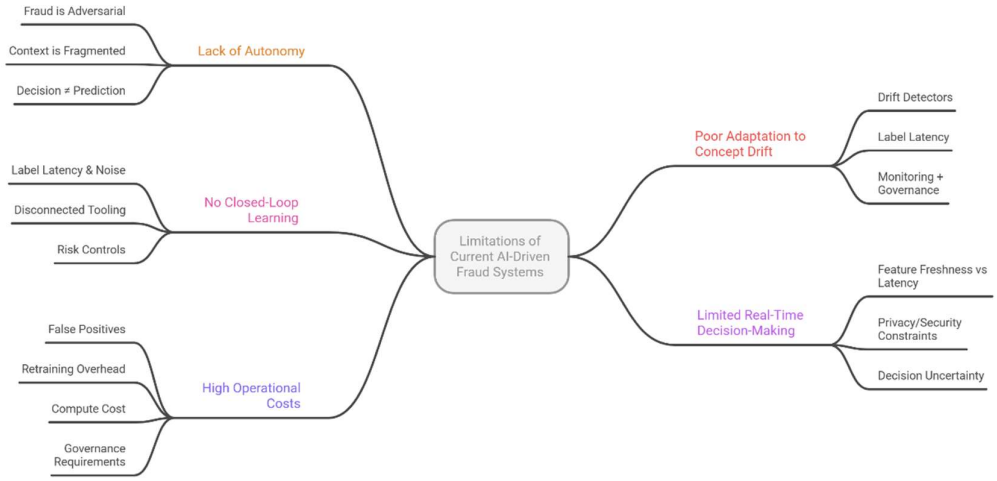

Figure 3: Limitations of current AI-Driven Fraud detection models

Limitation Area	Core Issue	Practical Impact in Production	Analytical Insight
Lack of Autonomy	Models act as scorers, not decision-makers	Decisions rely on rules + analysts; slow and inconsistent actions	High accuracy in lab, but AI becomes only one weak link in long workflows
	No action planning or coordination	Cannot choose next step (block, step-up, KYC, investigate)	Prediction ≠ decision; static boundaries are easy to evade
	Fragmented context	Cannot autonomously fetch graphs, device risk, history	Context loss reduces real-world effectiveness
Poor Drift Adaptation	Concept drift (global & local)	Model fails silently in specific segments	Model may look stable overall but fail badly in pockets
	Weak drift benchmarks	Trial-and-error deployment	No standard way to compare recovery speed or safety
	Delayed labels	Online learning impractical	Models score with outdated decision boundaries
No Closed-Loop Learning	Broken feedback loop	Analyst decisions not reused effectively	Learning lags far behind fraud evolution

	Label latency & noise	Weak supervision	"Not fraud" ≠ "legitimate"
	Risk governance friction	Continuous learning avoided	Stability prioritized over adaptability
Limited Real-Time Capability	End-to-end latency constraints	Rich features too slow for strict SLAs	Real-time ≠ fast inference alone
	Feature freshness vs speed	Tradeoff between context and latency	Near real-time replaces true real-time
	No utility optimization	Probability ≠ optimal action	Models do not optimize loss vs friction
High Operational Cost	Alert volume	Increased analyst workload	Model metrics ignore downstream cost
	Monitoring & retraining overhead	Continuous engineering burden	Multi-product stacks amplify cost
	Advanced model compute	GNNs / DL expensive at scale	Graph refresh & inference cost matter
	Governance & audit	Slower iteration	Compliance dominates agility

Table 7 : Limitations of current AI-Driven Fraud detection models

While current AI-driven fraud systems achieve strong prediction accuracy, their limitations become clear in real operational settings. Consider a common account takeover (ATO) scenario in a real-time payment system. A machine learning model flags a sudden change in device and transaction behavior and assigns a high fraud risk score. However, the system does not autonomously decide the next action. Instead, the decision is routed through static rules and manual review queues. In a high-volume payment environment, this delay can be costly. For example, if even 0.1% of legitimate transactions are falsely flagged in a system processing millions of daily transactions, thousands of customers may experience payment failures. This results in increased call-center workload, delayed investigations, and customer dissatisfaction. At the same time, genuinely fraudulent transactions may proceed during review delays, leading to direct financial loss. This example highlights a critical gap: current systems predict risk but lack the ability to act autonomously, gather additional evidence, or adjust response strategies in real time. As transaction volumes grow, these operational inefficiencies scale rapidly, underscoring the need for fraud detection systems that can combine prediction with timely and context-aware decision-making.

4. Agentic AI vs. Traditional AI: A Comparative Analysis

Over the past decade, fraud detection in banking has evolved significantly. Early rule-based and feature-engineered machine learning systems have been augmented by deep learning models, sequential architectures, graph neural networks for collusion detection, and privacy-aware approaches such as federated learning combined with explainable AI. Recent research on agentic AI proposes a

different paradigm. Instead of treating fraud detection as a sequence of disconnected steps around a classifier, agentic systems frame it as a closed-loop decision process. Such systems are designed to plan actions, use tools, coordinate specialized sub-agents, and adapt continuously, while maintaining governance, auditability, and compliance. This shift reflects growing recognition that effective fraud management requires more than accurate predictions—it requires operational decision intelligence.

Traditional AI vs. Agentic AI: System-Level Comparison

Dimension	Traditional AI Systems	Agentic AI Systems
Primary objective	Accurate prediction or risk scoring	End-to-end decision and response optimization
Architecture	Single model or fixed pipeline	Multi-agent orchestration with tool use and state
Evidence handling	Limited to internal features	Explicit evidence gathering and context enrichment
Concept drift	Periodic retraining, delayed response	Continuous monitoring with adaptive actions
Explainability	Post-hoc XAI (e.g., SHAP, LIME)	Reasoned decisions with evidence trails and logs
Fraud rings	Improved via GNNs	Graph tools combined with coordinated investigation
False positives	Often high due to static thresholds	Reduced via context-aware triage and routing
Compliance & governance	Largely external and manual	Built-in guardrails, traceability, and auditability
Operational focus	Alert generation	Case handling, escalation, remediation, learning

Figure 8: Traditional AI vs. Agentic AI: System-Level Comparison

5. Open Challenges and Research Directions

Despite substantial progress in machine learning, deep learning, graph neural networks, and explainable AI, fraud detection in banking continues to face structural and operational limitations. The emergence of Agentic AI represents a shift from passive prediction to autonomous, goal-driven decision-making, enabling systems to reason, plan, and act in real time. However, this shift introduces new challenges that must be resolved before deployment in regulated, high-risk financial environments [43].

Category	Primary Challenge	Recommended Solution
Trust & Governance	Accountability and auditability	Governance-by-design, audit trails
Ethics & Regulation	Bias and transparency	Fairness-aware learning and continuous monitoring
Scalability	Latency and coordination overhead	Event-driven, hierarchical architectures
Security	Agent communication attacks	Zero-trust and cryptographic protocols
Human Oversight	Maintaining accountability	Adaptive HITL frameworks
Open Challenges	Drift, benchmarking, legacy integration, efficiency	Standardized evaluation and adaptive architectures

Table 10 : Challenges with possible solutions

6. Case Studies and Industry Adoption

Over the past five years, financial institutions have increasingly transitioned from traditional rule-based fraud detection systems to advanced AI-driven frameworks. Academic and industry-oriented research published in high-quality journals (Q1–Q3) demonstrates that real-world adoption of artificial intelligence has significantly improved fraud detection accuracy, reduced false positives, and enhanced operational efficiency. This section reviews representative case studies across different financial domains, highlighting practical deployments, achieved outcomes, and observed limitations. Figure shows agentic AI driven fraud detection in finance domain.

Domain	Institution Type	Core Techniques	Primary Strengths	Operational Gaps	Comparative Insight
Credit Card Fraud	Commercial Bank	XGBoost, LSTM	Significant reduction in false positives; improved recall on sequential fraud	Frequent manual retraining; delayed drift response	Supervised ML improves precision but remains dependent on offline retraining cycles
Digital Payments	National Payment Network	Ensemble ML, Streaming Models	Real-time transaction scoring at scale	Concept drift handling is reactive	Streaming ML enables low latency but lacks proactive adaptation

Retail Banking	Retail Bank	DNN + SHAP	Regulatory-friendly explanations	Explainability applied post hoc	Current XAI supports compliance but is weakly coupled to decision logic
AML & Ring Fraud	Large Financial Institution	Graph Neural Networks (GNNs)	Detection of coordinated fraud and hidden relationships	High computational cost; limited real-time scalability	Graph models excel at structural fraud but face deployment constraints
Autonomous Fraud Systems	Pilot / Sandbox Deployments	Multi-Agent AI, Reinforcement Learning	Adaptive learning and closed-loop decision-making	Limited governance maturity; early adoption	Agentic systems shift fraud detection from prediction to decision-making

Table 9: Case studies for Industry Application

7. Agentic AI solution for fraud management in Banking

The figure 4 presents a multi-agent AI architecture designed to support financial risk intelligence in banking and e-commerce environments. At the center of the architecture is the Orchestrator or Coordination Agent, which acts as the supervisory component responsible for managing communication among all specialized agents. Rather than performing risk analysis itself, it distributes tasks, coordinates workflows, resolves conflicts between recommendations, and ensures that all agents operate as a unified system.

The operational layer consists of several domain-specific agents that focus on different aspects of risk management. The Credit Scoring Agent evaluates customer creditworthiness and estimates the likelihood of default by generating risk scores and credit grades. Alongside it, the Fraud Detection Agent continuously analyzes transactions and customer behavior to identify suspicious activities and potential fraud attempts. The Operational Risk Agent monitors internal business processes, systems, and operational events to detect weaknesses or disruptions that could impact organizational performance. The AML and Compliance Agent focuses on regulatory requirements by monitoring anti-money laundering activities, Know Your Customer obligations, and compliance-related risks.

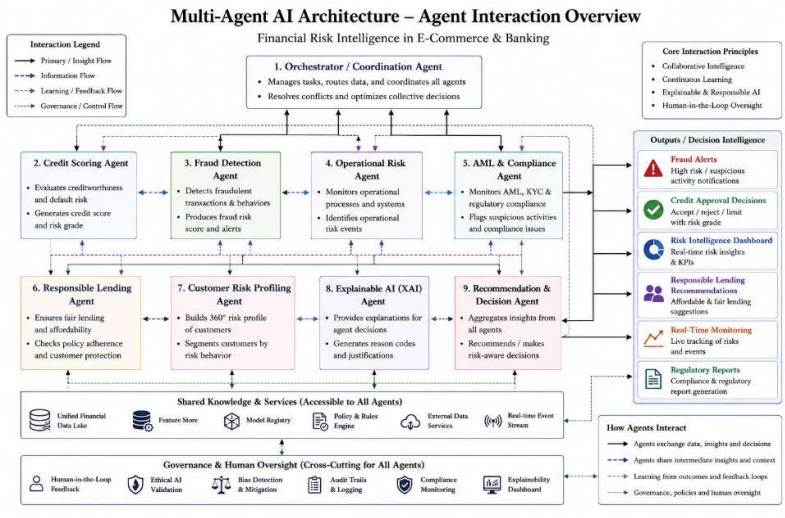

Figure 4: Agentic AI Solution for Banking Domain Application

Supporting these primary agents are additional intelligence layers that enrich decision-making. The Responsible Lending Agent ensures that lending practices remain fair, transparent, and aligned with regulatory and ethical requirements. The Customer Risk Profiling Agent develops a comprehensive view of customer behavior by combining information from multiple sources and identifying patterns that indicate different levels of risk. The Explainable AI Agent provides transparency by generating understandable explanations for decisions made by the system, helping regulators, auditors, and business users understand why specific outcomes were produced. The Recommendation and Decision Agent serves as the final analytical layer, consolidating insights from all other agents and producing risk-aware recommendations and actions. The architecture is built upon a shared knowledge and services layer that provides common resources to every agent. This layer includes a unified financial data lake, feature stores, model registries, policy engines, external data services, and real-time event streams. By accessing the same information sources, agents maintain consistency in their analyses and decisions while avoiding data silos.

A key characteristic of the framework is the continuous interaction between agents. Information flows horizontally across the architecture, allowing agents to exchange insights, risk indicators, and contextual information. Learning and feedback mechanisms enable agents to improve their performance over time by incorporating outcomes from previous decisions. Governance and control flows ensure that all activities remain aligned with organizational policies and regulatory expectations. At the foundation of the architecture lies a governance and human oversight layer. This layer incorporates human-in-the-loop feedback, ethical AI validation, bias detection and mitigation, audit trails, compliance monitoring, and explainability dashboards. These capabilities ensure that autonomous decision-making remains transparent, accountable, and subject to appropriate supervision.

The collective output of the architecture is transformed into actionable business intelligence. The system generates fraud alerts, credit approval decisions, risk intelligence dashboards, responsible lending recommendations, real-time monitoring insights, and regulatory reports. By integrating specialized agents, shared knowledge resources, and governance controls, the architecture creates a collaborative environment where multiple AI agents work together to deliver accurate, explainable, and compliant financial risk management across banking and e-commerce operations.

7. Conclusion

This review presented a comprehensive analysis of fraud detection techniques in banking and financial systems, highlighting their evolution from traditional rule-based mechanisms to advanced AI-driven approaches and, more recently, to Agentic AI paradigms. Conventional statistical and machine learning models laid the foundation for automated fraud detection but often struggled with issues such as static decision rules, high false-positive rates, and limited adaptability to evolving fraud patterns. Deep learning and graph-based models significantly improved detection accuracy by capturing complex temporal, behavioral, and relational patterns in financial transactions. However, these approaches still largely operate as isolated predictive systems without true autonomy or continuous learning capabilities. The emergence of Agentic AI marks a paradigm shift in financial fraud detection. By enabling autonomous decision-making, reasoning, and action through closed-loop feedback systems, Agentic AI addresses key limitations of existing models, including concept drift, delayed response, and limited explainability. Multi-agent collaboration further enhances system robustness by integrating detection, investigation, learning, and compliance functions..

References

I. Vorobyev and A. Krivitskaya, "Reducing false positives in bank anti-fraud systems based on rule induction in distributed tree-based models," Computers & Security, vol. 120, p. 102786, 2022. https://doi.org/10.1016/j.cose.2022.102786

Ahmed M, Ansar K, Muckley CB, Khan A, Anjum A, Talha M. A semantic rule based digital fraud detection. PeerJ Comput Sci. 2021 Aug 3;7:e649. doi: 10.7717/peerj-cs.649. PMID: 34435097; PMCID: PMC8356649.

B. Baesens, "Data engineering for fraud detection," Decision Support Systems, p. 113492, 2021. https://doi.org/10.1016/j.dss.2021.113492

J. S. Allen et al., "Bayesian Quickest Detection of Credit Card Fraud," Bayesian Analysis, 2022. https://doi.org/10.1214/20-BA1254

O. Ogundile et al., "Credit Card Fraud: Analysis of Feature Extraction Techniques for Ensemble Hidden Markov Model Prediction Approach," Applied Sciences, vol. 14, no. 16, p. 7389, 2024.

https://doi.org/10.3390/app14167389

A. Razaque et al., "Credit Card-Not-Present Fraud Detection and Prevention Using Big Data Analytics Algorithms," Applied Sciences, vol. 13, no. 1, p. 57, 2022. https://doi.org/10.3390/app13010057

S. Dalal et al., "Predicting Fraud in Financial Payment Services through Optimized Hyper-Parameter-Tuned XGBoost Model," Mathematics, vol. 10, no. 24, p. 4679, 2022.

https://doi.org/10.3390/math10244679

D. Cirqueira et al., "Towards Design Principles for User-Centric Explainable AI in Fraud Detection," (conference/book chapter listing), 2021.

https://doi.org/10.1007/978-3-030-77772-2_2

J. K. Afriyie et al., "A supervised machine learning algorithm for detecting and predicting fraudulent credit card transactions," (Sciencedirect-indexed article), 2023

Y. Chen, C. Zhao, Y. Xu, C. Nie, and Y. Zhang,

"Deep learning in financial fraud detection: innovations, challenges, and applications,"

Data Science and Management, in press, Aug. 2025, doi: 10.1016/j.dsm.2025.08.002.

S. Motie and B. Raahemi,"Financial fraud detection using graph neural networks: a systematic review,"

Expert Systems with Applications, vol. 240, Dec. 2024, doi: 10.1016/j.eswa.2023.122156.

S. K. Tagbo and A. F. Adekoya,"A systematic literature review of machine learning techniques in financial fraud prevention and detection," International Journal of Financial Research, Aug. 2025 (preprint available; DOI may exist in final publication).

M. Dal Pozzolo, O. Bontempi, and G. Snoeck,"Adversarial drift detection in real-world fraud detection systems,"

IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 6, pp. 2139–2150, Jun. 2020, doi: 10.1109/TNNLS.2019.2931996.

F. Grandi, M. Re, and L. Rossi, "A semantic rule-based approach for fraud detection in digital payments," Applied Intelligence, vol. 52, pp. 13743–13761, 2022, doi: 10.1007/s10489-022-03347-1.

A. Dal Pozzolo, G. Bontempi, and O. Snoeck, "Calibrating probability with undersampling for unbalanced classification," IEEE Symposium on Computational Intelligence and Data Mining, 2019, doi: 10.1109/CIDM.2019.00020.

Y. Xie and D. Siegmund, "Sequential analysis for online fraud detection,"

Annals of Statistics, vol. 41, no. 5, pp. 2375–2400, 2013, doi: 10.1214/13-AOS1156.

O. Ogundile, O. Babalola, A. Ogunbanwo, O. Ogundile, and V. Balyan, "Credit Card Fraud: Analysis of Feature Extraction Techniques for Ensemble Hidden Markov Model Prediction Approach," Applied Sciences, vol. 14, no. 16, 2024, Art. no. 7389,

doi: 10.3390/app14167389.

M. Z. H. George, M. K. Alam, and M. T. Hasan, "Machine Learning for Fraud Detection in Digital Banking: A Systematic Literature Review,"

arXiv preprint, 2025, doi: 10.48550/arXiv.2510.05167

Y. Chen, C. Zhao, Y. Xu, and C. Nie, "Year-over-Year Developments in Financial Fraud Detection via Deep Learning: A Systematic Literature Review,"

arXiv preprint, 2025, doi: 10.48550/arXiv.2502.00201.

G. Charizanos, "An online fuzzy fraud detection framework for credit card transactions," Expert Systems with Applications, vol. 234, 2024, Art. no. 120685,

doi: 10.1016/j.eswa.2024.120685

Ileberi, E., Sun, Y. & Wang, Z. A machine learning based credit card fraud detection using the GA algorithm for feature selection. J Big Data 9, 24 (2022). https://doi.org/10.1186/s40537-022-00573-8.

N. Baisholan, J. E. Dietz, S. Gnatyuk, M. Turdalyuly, E. T. Matson, and K. Baisholanova, "FraudX AI: An Interpretable Machine Learning Framework for Credit Card Fraud Detection on Imbalanced Datasets," Computers, vol. 14, no. 4, Art. no. 120, Apr. 2025, doi: 10.3390/computers14040120.

S. Iqbal, "Interpretable Ensemble Learning Models for Credit Card Fraud Detection using Random Forest and XGBoost with SHAP and LIME," Appl. Sci., vol. 15, no. 22, Art. no. 12073, Nov. 2025, doi: 10.3390/app152212073

P. Hajek, M. Z. Abedin, and U. Sivarajah, "Fraud detection in mobile payment systems using an XGBoost-based framework," Information Systems Frontiers, vol. 25, no. 5, pp. 1985–2003, 2023, doi: 10.1007/s10796-022-10346-6.

B. Dalal, "Hyperparameter-tuned XGBoost for payment fraud prediction,"

Mathematics, vol. 10, no. 24, p. 4679, 2022, doi: 10.3390/math10244679.

A. Dal Pozzolo, G. Bontempi, O. Snoeck, and G. Snoeck, "Calibrating probability with undersampling for unbalanced classification,"

in Proc. IEEE Symp. Comput. Intell. Data Mining, 2015, pp. 159–166,

doi: 10.1109/SSCI.2015.33.

J. Davis and M. Goadrich, "The relationship between precision-recall and ROC curves,"

in Proc. 23rd Int. Conf. Machine Learning (ICML), 2006, pp. 233–240,doi: 10.1145/1143844.1143874.

L. Hernandez Aros, L. X. Bustamante Molano, F. Gutierrez-Portela, et al.,

"Financial fraud detection through the application of machine learning techniques: a literature review,"

Humanities and Social Sciences Communications, vol. 11, p. 1130, Sep. 2024, doi: 10.1057/s41599-024-03606-0

X. Li, Y. Peng, X. Sun, Y. Duan, Z. Fang, and T. Tang,"Unsupervised detection of fraudulent transactions in e-commerce using contrastive learning,"

arXiv, Mar. 24, 2025.

F. Moradi, M. Tarif, and M. Homaei, "Semi-supervised supply chain fraud detection with unsupervised pre-filtering," arXiv, Aug. 7, 2025.

E. F. Agyemang, "Anomaly detection using unsupervised machine learning techniques," ScienceDirect, 2024.

Hajek, P., Abedin, M.Z. & Sivarajah, U. Fraud Detection in Mobile Payment Systems using an XGBoost-based Framework. Inf Syst Front 25, 1985–2003 (2023). https://doi.org/10.1007/s10796-022-10346-6

Ahmad H, Kasasbeh B, Aldabaybah B, Rawashdeh E. Class balancing framework for credit card fraud detection based on clustering and similarity-based selection (SBS). Int J Inf Technol. 2023;15(1):325-333. doi: 10.1007/s41870-022-00987-w. Epub 2022 Jun 21. PMID: 35757149; PMCID: PMC9209320.

Du, Haichao, Li Lv, An Guo, and Hongliang Wang. 2023. "AutoEncoder and LightGBM for Credit Card Fraud Detection Problems" Symmetry 15, no. 4: 870. https://doi.org/10.3390/sym15040870.

Shanaa M and Abdallah S. A Hybrid Anomaly Detection Framework Combining Supervised and Unsupervised Learning for Credit Card Fraud Detection [version 2; peer review: 1 approved with reservations, 1 not approved]. F1000Research 2025, 14:664 (https://doi.org/10.12688/f1000research.166350.2)

Bulla, C., Birje, M.N. Improved data-driven root cause analysis in fog computing environment. J Reliable Intell Environ 8, 359–377 (2022). https://doi.org/10.1007/s40860-021-00158-x

I. Benchaji, S. Douzi, B. El Ouahidi, and J. Jaafari, "Enhanced credit card fraud detection based on attention mechanism and LSTM deep model," Journal of Big Data, vol. 8, art. no. 151, 2021.

S. Jiang, R. Dong, J. Wang, and M. Xia, "Credit Card Fraud Detection Based on Unsupervised Attentional Anomaly Detection Network," Systems, vol. 11, no. 6, art. no. 305, 2023.

Forough and S. Momtazi, "Ensemble of deep sequential models for credit card fraud detection," Applied Soft Computing, vol. 99, art. no. 106883, 2021.

Y. Wu, "A Deep Learning Method of Credit Card Fraud Detection Based on Continuous-Coupled Neural Networks," Mathematics, vol. 13, no. 5, art. no. 819, 2025.

Y. Chen, C. Zhao, Y. Xu, C. Nie, and Y. Zhang,"Deep learning in financial fraud detection: Innovations, challenges, and applications," Data Science and Management, in press, available online 20 Aug. 2025, doi: 10.1016/j.dsm.2025.08.002.

Hooi, Bryan, et al. "Graph-based fraud detection in the face of camouflage." ACM Transactions on Knowledge Discovery from Data (TKDD) 11.4 (2017): 1-26.

D. Cheng, Y. Zhang, F. Shi, and Y. Liu, "Graph neural networks for financial fraud detection: A review," Expert Systems with Applications, vol. 234, 2024, doi: 10.1016/j.eswa.2023.122156.

P. Hajek, M. Z. Abedin, and U. Sivarajah, "Fraud detection in mobile payment systems using graph-based learning," Information Systems Frontiers, vol. 25, no. 5, pp. 1985–2003, 2023, doi: 10.1007/s10796-022-10346-6.

M. Du, Y. Ding, and X. Wang, "Autoencoder-enhanced graph learning for financial fraud detection," Symmetry, vol. 15, no. 3, 2023, doi: 10.3390/sym15030567.

A. Benchaji, M. Douzi, and B. Bouikhalene, "Sequential fraud detection using attention-based LSTM networks," Applied Soft Computing, vol. 109, 2021, doi: 10.1016/j.asoc.2021.107540.

D. Forough and S. Momtazi, "Deep sequential models for credit card fraud detection," Applied Soft Computing, vol. 104, 2021, doi: 10.1016/j.asoc.2021.107186.

S. Shanaa and S. Abdallah, "Hybrid anomaly and graph-based fraud detection using attention mechanisms," F1000Research, vol. 14, 2025, doi: 10.12688/f1000research.166350.1.

Y. Wang, Z. Zhang, and J. Tang, "Heterogeneous graph neural networks for supply chain finance fraud detection," Decision Support Systems, vol. 176, 2024, doi: 10.1016/j.dss.2023.113789.

L. Wu, P. Cui, J. Pei, and L. Zhao, "Graph neural networks in blockchain fraud detection," in Proc. MLG Workshop, 2023..