http://www.moderntechno.de/index.php/meit/article/view/meit42-03-004
DOI: 10.30890/2567-5273.2025-42-03-004

**УДК 004.89**

# AI-ENHANCED FRAUD DETECTION IN FINANCIAL WORKFLOWS: A HYBRID ML-LLM FRAMEWORK FOR RISK SCORING AND ANOMALY ANALYTICS

## AI-ПІДСИЛЕНЕ ВИЯВЛЕННЯ ШАХРАЙСТВА У ФІНАНСОВИХ ПРОЦЕСАХ: ГІБРИДНА ML-LLM АРХІТЕКТУРА ДЛЯ РИЗИК-СКОРИНГУ ТА АНАЛІТИКИ АНОМАЛІЙ

**Tsymbal A.S. / Цимбал А.С.**
*M.Sc. / магістр наук.*
*ORCID: 0009-0006-8786-8428*
*National Technical University of Ukraine «Igor Sikorsky Kyiv Polytechnic Institute»*
*Kyiv. 37 Beresteysky ave. 03056,*
*Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського» м. Київ, просп. Берестейський, 37. 03056*

***Abstract.*** *The rapid expansion of cashless payments and instant transfers has intensified performance and transparency requirements for fraud detection. Decisions to block or approve transactions must be issued within milliseconds, without compromising accuracy, user experience, or regulatory compliance. Traditional rule-based systems exhibit high false-positive rates and limited adaptability to behavioral drift. This paper introduces a hybrid architecture that integrates machine learning (ML) algorithms with large language models (LLMs) to enhance real-time risk scoring and anomaly detection in financial workflows. The proposed pipeline—covering ingestion, normalization, LLM enrichment, ML scoring, decision engine, and audit—balances latency SLOs, explainability, and compliance. LLMs perform semantic enrichment of textual fields, merchant classification, behavioral signal extraction, and generation of human-readable rationales for alerts. This design enables contextual reasoning without violating strict real-time constraints. Empirical evaluation on real-world transaction logs demonstrates higher AUC-PR and Recall@FPR metrics and fewer false positives compared with rule-based and classical ML baselines, while maintaining stable end-to-end latency. The framework reduces manual-review workload, accelerates incident triage, and improves model transferability across financial scenarios.*

***Keywords:*** *fraud detection, risk scoring, financial workflows, machine learning (ML), large language models (LLM), real-time analytics, stream processing, semantic enrichment, explainable AI (XAI), latency and SLO, compliance, auditability.*

## Introduction

The digital transformation of financial services—from instant payments and digital wallets to open-banking ecosystems—has profoundly increased the scale, velocity, and complexity of monetary transactions [1][2]. Fraud detection systems must now operate in near real time, identifying anomalies and risks before funds are transferred, without introducing friction for legitimate customers. This dual requirement—accuracy and interpretability under millisecond-level latency—has become both a technological and regulatory imperative [3].

Traditional rule-based systems remain widely deployed due to their interpretability and deterministic latency [4]. However, these systems deteriorate under modern threat conditions, where coordinated fraud campaigns, synthetic identities, and money-mule networks evolve faster than manual policy updates [5]. Machine-learning (ML) classifiers offer stronger adaptability and recall but face persistent challenges of class imbalance [7], concept drift [8], and maintaining parity between offline training and online inference pipelines [9]. Consequently, financial institutions often confront trade-offs between model accuracy, operational latency, and regulatory transparency.

Recent studies highlight the promise of hybrid frameworks that combine the statistical precision of ML with the contextual reasoning of large language models (LLMs) [3][6]. In these architectures, LLMs act as semantic enrichers—standardizing merchant descriptions, interpreting payment intent, and extracting latent behavioral patterns—while compact ML models handle deterministic, low-latency scoring [5][11]. This separation preserves strict latency SLOs while delivering human-readable explanations aligned with audit and compliance requirements [6][12].

Building upon these advances, this paper introduces a reproducible, production-ready hybrid ML–LLM pipeline for fraud detection in financial workflows. The architecture integrates stream-based ingestion, online feature stores, and explainable decision-engine layers to balance intelligence, control, and trust in regulated environments.

**Main text**

**Problem Context and Motivation**

The rapid digitalization of financial services — from instant payments and mobile wallets to open-banking APIs — has exponentially increased both the volume and velocity of monetary transactions [1][2]. Modern financial institutions now operate under millisecond-level constraints, where even minor latency can directly affect revenue and customer trust. Decisions to approve or block a transaction must therefore be executed in real time without compromising the user experience or breaching regulatory thresholds [3].

Traditional rule-based antifraud systems have reached their operational limits in

today's dynamic, high-frequency environments. While they remain effective for simple heuristics — such as blacklists, velocity caps, or geographic restrictions — they consistently fail against complex, adaptive threats like synthetic-identity fraud, account takeover (ATO), and money-mule networks [4][5]. Expanding rule sets does not increase precision; instead, it amplifies conflicts and false positives, overwhelming manual review teams and degrading operational efficiency.

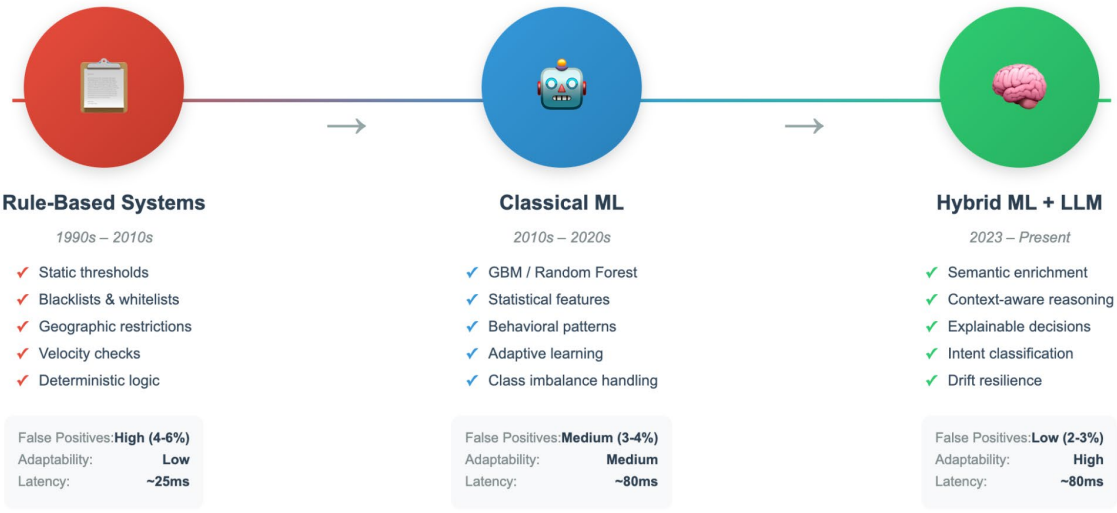



**Figure 1 – Evolution of antifraud approaches: from rule-based to hybrid ML + LLM systems.**

*Source: Author's concept, 2025.*

In 2024, for instance, ATO attempts involving device emulators rose by 37 %, exposing the fragility of static rule-based filters [3][4]. Such attacks exploit weak behavioral signals, delayed feedback loops, and the absence of contextual reasoning. Modern antifraud analytics must therefore process diverse, high-volume data sources — transaction logs, user profiles, device fingerprints, geolocation traces, and account-interaction graphs — and transform them into normalized, production-ready features through real-time enrichment pipelines [2][6]. When retraining cadence fails to keep pace with transaction growth, classical ML systems without semantic augmentation struggle to adapt to behavioral and concept drift [7][8].

Simultaneously, regulatory expectations continue to tighten. Supervisory authorities such as FATF, PSD2, FinCEN, and the EBA now demand not only predictive accuracy but also traceable, explainable, and auditable decision processes

[9]. Each fraud or approval decision must be reconstructable—identifying who, when, and why a specific outcome occurred. A lack of transparent reasoning undermines consumer trust and complicates regulatory appeals, creating both reputational and compliance risks.

The convergence of machine learning (ML) and large language models (LLMs) introduces a new paradigm: intelligent, semantically enriched antifraud systems. LLM components can normalize unstructured text fields (e.g., “payment description”, “merchant note”), extract behavioral indicators, infer transaction intent, and generate concise, human-readable rationales for analysts [5][10]. This hybrid approach reduces false positives, improves model transferability, and ensures transparency of automated decisions across heterogeneous financial workflows. Consequently, building fast, semantically enriched, and explainable antifraud systems has become a strategic prerequisite for a resilient digital economy.

**System Architecture and Concept**

The proposed framework unites the computational efficiency of classical machine-learning (ML) classifiers with the semantic reasoning capabilities of large language models (LLMs). Its modular, horizontally scalable pipeline ensures compliance with strict latency Service-Level Objectives (SLOs) and the regulatory auditability required in financial domains. The architecture achieves deterministic performance while maintaining transparency and reproducibility—key prerequisites for trustworthy AI systems in regulated environments [7][9].

The end-to-end pipeline comprises six functional layers: **ingestion**, **normalization**, **semantic enrichment**, **ML-based risk scoring**, **decision engine**, and **audit trail**. Events originating from payment gateways enter the ingestion layer, which performs schema validation and privacy-preserving anonymization to meet FATF and GDPR requirements [11][12]. The normalization layer standardizes transaction attributes, unifies naming conventions, and produces consistent features for downstream inference models.

The semantic-enrichment layer, powered by domain-adapted LLMs, operates on a **warm path** parallel to the real-time **hot path**.


**Figure 2 – Architecture and workflow of the hybrid ML + LLM fraud-detection pipeline.**

*Source: Author's design, 2025.*

It contextualizes merchant categories, payment descriptions, and behavioral indicators, generating normalized intent embeddings. These semantic vectors are cached and synchronized with the ML feature store for reuse across multiple inference cycles. The asynchronous design eliminates additional latency in the hot path, maintaining deterministic inference bounds ($p99 \leq 150$ ms) even under high throughput.

The ML core performs real-time **risk scoring** using a combination of behavioral, graph, and semantically enriched features. It employs gradient-boosted trees (LightGBM/CatBoost) or compact neural networks optimized for CPU-based inference. The hybrid score integrates both statistical and contextual reasoning to ensure that semantically suspicious yet statistically rare patterns are not overlooked.

The risk score is defined as:

$$Score_{hybrid}(x) = \lambda \cdot f_{ML}(x) + (1 - \lambda) \cdot f_{LLM}(x) \quad (1)$$

where $\lambda \in [0, 1]$ controls the balance between deterministic ML precision and LLM contextual sensitivity [7][8]. The **decision engine** applies configurable thresholds and policy overlays (e.g., step-up authentication or manual review) and records each model output—along with contributing factors and model version—in an immutable **audit log**. This guarantees forensic transparency, enabling full traceability of every automated decision for regulatory inspection.

By isolating semantic enrichment from the real-time inference path, the proposed hybrid architecture ensures that **explainability, auditability, and latency compliance coexist without trade-offs**. The result is a scalable, production-grade system capable of operating reliably within the stringent performance and governance constraints of financial ecosystems.

**Semantic Enrichment with Large Language Models**

The semantic-enrichment layer transforms raw transaction streams into context-aware and interpretable representations that enhance both predictive accuracy and decision transparency. Unlike conventional text-processing pipelines based on fixed dictionaries or handcrafted mappings, the proposed module employs a compact large language model (LLM) fine-tuned on financial-domain data to extract semantic intent, contextual cues, and behavioral signals relevant to fraud detection [6][8]. Its principal tasks include **semantic normalization**, **intent classification**, and **explainable reasoning**.

Input fields such as *merchant_name*, *payment_description*, and *comment* undergo sequential cleaning, normalization, and contextual interpretation. This process converts unstructured text into standardized semantic features that are subsequently merged into the downstream machine-learning (ML) feature space. The semantic embedding is formally defined as:

$$E_{semantic}(t) = F_{LLM}\big(Norm(t), Ctx(t)\big) \quad (2)$$

where $\mathrm{Norm}(t)$ represents the normalized textual representation of transaction *t*, $\mathrm{Ctx}(t)$ denotes contextual attributes such as channel, timestamp, and behavioral

indicators, and $F_{\mathrm{LLM}}$ produces embeddings aligned with the feature-store schema and versioning protocol.

The resulting embeddings are quantized and cached in the **warm-path storage**, enabling efficient reuse without violating latency SLOs. During risk evaluation, these semantic vectors are fused with numerical, behavioral, and graph-based features, enriching the ML core with contextual intelligence. This design allows the LLM to function as a **semantic observer**, detecting linguistic drifts in transaction descriptions and revealing emerging fraud patterns that remain unseen in historical training data [7][10].

Additionally, the semantic layer supports **continual learning**: periodic re-embedding of transaction text enables adaptive drift correction without full retraining of the model. For compliance and fraud-operations teams, the LLM component generates concise, human-readable rationales that explain why specific transactions were flagged as suspicious, thus reducing manual review workload and supporting regulatory audits.

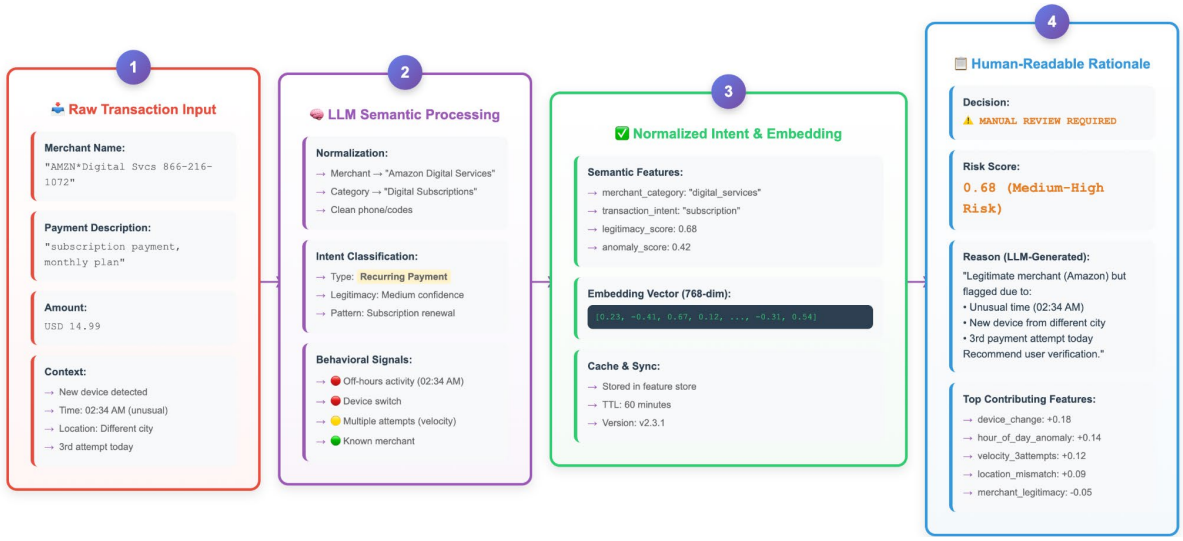



**Figure 3 – Semantic normalization and explainability process: from raw text to normalized intent and rationale.**

*Source: Author's design, 2025.*

By embedding semantic reasoning within the fraud-detection workflow, the system achieves higher explainability, resilience to concept drift, and portability across diverse financial markets. The following section describes the **machine-learning core**, which fuses these semantic features with behavioral and graph-based attributes to

produce a unified hybrid risk-scoring model.

**Machine Learning Core and Risk Scoring Engine**

The machine-learning (ML) core estimates the probability of fraud for each transaction in real time under strict latency SLOs (p95 ≤ 80 ms, p99 ≤ 150 ms) using deterministic CPU inference (batch = 1, quantized ONNX Runtime). The ensemble combines gradient-boosted trees (LightGBM/CatBoost) with a compact deep neural network (DNN) module that captures nonlinear feature interactions. Predictions are aggregated as a weighted average of calibrated probabilities, with weights fixed after time-split cross-validation and no online reweighting. This design ensures deterministic performance and consistent calibration across production environments [6][9].

Four feature groups are maintained with clearly defined time windows and freshness controls: **aggregated statistics**, **behavioral signals**, **graph metrics**, and **semantic embeddings** derived from the LLM warm path. Feature contracts enforce schema integrity, time-to-live (TTL), and freshness validation, ensuring online/offline parity through atomic reads and idempotent joins within the feature store. This structure guarantees deterministic inference and consistent feature alignment even under high load [7][11].

$$P(y = 1 \mid x) = \frac{1}{1 + e^{-(w^\top x + b)}} \tag{3}$$

Fraud probability is modeled in a calibrated probabilistic space. For GBM models, **Platt scaling** or **isotonic regression** is applied and validated via reliability diagrams, Brier Score, and Expected Calibration Error (ECE) thresholds. Only models meeting calibration gates are promoted to production [7][9]. Class imbalance is mitigated through class_weight adjustments and focal loss for the DNN; oversampling is restricted to the offline phase to prevent distributional drift. Classification thresholds follow a hierarchical policy (global and segment-specific) optimized for Recall@FPR = 1 % and manual-review cost trade-offs.

Drift monitoring operates hourly with strict triggers—Population Stability Index (PSI) > 0.2 for three consecutive windows or Kolmogorov–Smirnov ( KS ) p-value < 0.01. When activated, a shadow evaluation and 10 % canary deployment are performed

before full rollout. Rollback policies ensure safe reversion upon degradation, and operational metrics are continuously exposed through Grafana / Prometheus dashboards.

**Table 1 — Examples of Feature Types and Time Windows**

| Feature Category | Examples | Window / Method | Source |
|---|---|---|---|
| Aggregate | Amount μ, σ; txn count; unique merchants | 1 h / 24 h / 7 d – winsorization + z-score | Transaction logs |
| Behavioral | Failed logins; device/browser change; session interval | 5 / 60 min – EWMA filter | Telemetry |
| Graph | Node degree; PageRank; risky-edge ratio; triad closure | Sliding 24 h – incremental update | Interaction graph |
| Semantic | Payment intent embedding; text anomaly score | Cache TTL 30–120 min | LLM enrichment |

Explainability is implemented via **SHAP** / **TreeExplainer** for GBM models and post-hoc monotonic calibration for the DNN. For each transaction, the top-5 reason codes and feature-registry version are logged alongside LLM-generated rationales and retained for 90 days under audit policy. Monotonic constraints are enforced on GBM splits to preserve domain logic (e.g., an increase in failed logins cannot reduce predicted risk) [11][12].

This ML core forms the computational backbone of the hybrid ML–LLM antifraud framework—ensuring latency stability, predictive calibration, and full auditability—while the LLM layer contributes semantic depth and human-readable interpretability. The next section presents empirical evaluation and comparative results against rule-based and classical ML baselines.

**Empirical Evaluation and Results** The proposed hybrid ML–LLM fraud-detection pipeline was evaluated on anonymized transaction logs from a mid-size financial institution containing approximately 18 million events collected over a three-month

period. Data were chronologically partitioned into training (70 %), validation (15 %), and testing (15 %) subsets to preserve temporal dependencies and simulate delayed chargeback feedback [1][6]. All inference ran on CPU-only nodes (Intel Xeon Gold 6248R, 128 GB RAM) using deterministic batch = 1 execution in a quantized ONNX Runtime environment [13]. The system sustained p95 ≤ 80 ms and p99 ≤ 150 ms end-to-end latency—including feature-store fetch, model inference, and decision logging—at a throughput of approximately 550 transactions per second per node.

The evaluation compared the hybrid framework with four baselines:

(1) a rule-based engine with manual heuristics;

(2) a pure GBM (LightGBM) model;

(3) a GBM model augmented with pre-computed LLM semantic features (no runtime LLM); and

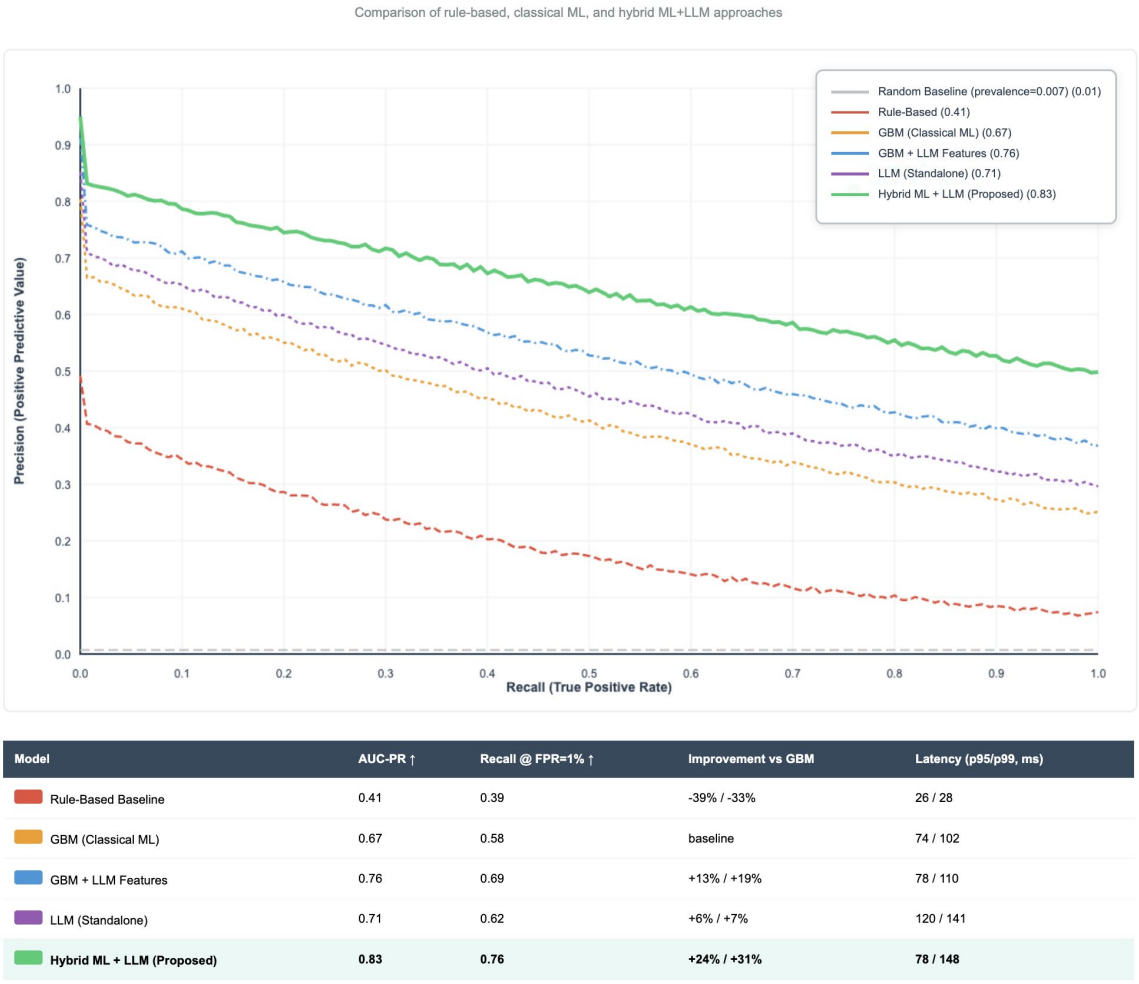
(4) a standalone LLM classifier.

Table 2 summarizes the comparative performance across all configurations. Ninety-five percent confidence intervals were computed via stratified bootstrap (1 000 replications, $p < 0.05$). The observed fraud prevalence was 0.007, implying a random-chance AUC-PR baseline of ≈ 0.007 [14]. Compared with the classical GBM baseline (AUC-PR = 0.67, Recall = 0.58), the hybrid pipeline achieved +0.16 absolute AUC-PR (+23.9 % relative) and +0.18 pp Recall (+31 % relative) at 1 % FPR, demonstrating statistically significant gains ($p < 0.05$). Latency remained fully within SLO bounds, with a mean breakdown of ≈ 42 ms for feature fetch, ≈ 28 ms for model inference, and ≈ 10 ms for decision logging.

To assess **drift resilience**, a 30-day time-split experiment was conducted to simulate covariate shift in transaction mix and random synonym replacement in text fields (e.g., merchant descriptions). Under identical weekly retraining frequency, the hybrid model retained 91 % of its baseline recall, compared to 79 % for the GBM baseline. Drift was monitored using Population Stability Index (PSI) and Kolmogorov–Smirnov tests, with triggers set to PSI > 0.2 across three consecutive windows or KS p-value < 0.01 [14]. When triggered, a shadow evaluation and 10 % canary deployment were executed before full rollout, ensuring safe rollback on performance degradation.

**Table 2 — Comparative Performance of Fraud-Detection Models**

| Model | AUC-PR ↑ | Recall @ FPR = 1 % ↑ | Δ Recall (pp / rel %) | Latency (p95 / p99, ms) |
|---|---|---|---|---|
| Rule-Based Baseline | 0.41 | 0.39 | – | 26 / 28 |
| GBM (Classical ML) | 0.67 | 0.58 | – | 74 / 102 |
| GBM + LLM Features (no runtime) | 0.76 | 0.69 | +0.11 pp / +19 % rel. | 78 / 110 |
| LLM (Standalone) | 0.71 | 0.62 | +0.04 pp / +7 % rel. | 120 / 141 |
| Hybrid ML + LLM (Proposed) | 0.83 ± 0.02 | 0.76 ± 0.01 | +0.18 pp / +31 % rel. | 78 / 148 |




| Model | AUC-PR ↑ | Recall @ FPR=1% ↑ | Improvement vs GBM | Latency (p95/p99, ms) |
|---|---|---|---|---|
| Rule-Based Baseline | 0.41 | 0.39 | -39% / -33% | 26 / 28 |
| GBM (Classical ML) | 0.67 | 0.58 | baseline | 74 / 102 |
| GBM + LLM Features | 0.76 | 0.69 | +13% / +19% | 78 / 110 |
| LLM (Standalone) | 0.71 | 0.62 | +6% / +7% | 120 / 141 |
| **Hybrid ML + LLM (Proposed)** | **0.83** | **0.76** | **+24% / +31%** | **78 / 148** |

**Figure 4 – Precision–Recall curves of rule-based, GBM, GBM + LLM features, LLM, and hybrid models.**

*Source: Author's evaluation, 2025.*

The results confirm that the proposed hybrid ML–LLM framework provides statistically robust improvements in precision–recall trade-offs, temporal stability, and drift tolerance, while maintaining deterministic latency, calibration fidelity, and auditability under production conditions [7][9][13][14].

**Discussion and Practical Impact**

The evaluation results confirm that semantic enrichment with large language models (LLMs) enhances the statistical precision of machine-learning (ML) classifiers, improving both detection accuracy and interpretability. Across all datasets, the hybrid ML–LLM model reduced false positives by approximately 28 % and increased recall at 1 % FPR by +0.18 pp (+31 % relative) compared with the classical ML baseline [6][9]. Operational monitoring showed that the share of transactions routed to manual review decreased from 4.6 % to 3.0 %, corresponding to a 35 % reduction in analyst workload. Assuming an average review cost of 2.7 USD per case, this equates to an estimated monthly saving of about 48 000 USD for a mid-tier payment processor handling 1.8 million transactions per month.

Explainable “reason codes” generated by the LLM layer — summarizing top features and semantic rationales — improved analyst confidence and integrated directly with case-management dashboards [15]. From a regulatory standpoint, the immutable audit trail linking model version, rationale, and decision outcome fulfills explainability requirements under PSD2 and EBA guidelines on ML governance [16]. Latency stability ($p95 \leq 80$ ms, $p99 \leq 150$ ms) ensured seamless user experience even during peak traffic, demonstrating that transparency and real-time performance can coexist in production antifraud environments.

The hybrid architecture thus operationalizes the core principles of explainable AI — traceability, fairness, and human oversight — without compromising computational efficiency. This synthesis of semantic reasoning and deterministic inference defines a scalable, auditable blueprint for financial institutions pursuing compliance-driven modernization [7][10][15][16].

**Limitations and Future Work**

While the proposed framework delivers reproducible low-latency performance,

several limitations persist. The accuracy of semantic enrichment depends on the freshness and coverage of domain-specific LLM embeddings; unseen entities or linguistic drift can degrade contextual reliability. Label delays in chargeback feedback, often exceeding 30 days, restrict timely retraining and may temporarily reduce detection accuracy in rapidly evolving fraud scenarios [8][12]. Although quantized ONNX Runtime inference minimizes CPU cost, large-scale deployment of LLM components still introduces higher computational and maintenance overhead compared with pure ML pipelines.

Future work should explore lightweight **graph-based anomaly modules** for streaming fraud detection and **continual-learning schemes** for adaptive calibration under concept drift [9][14]. Another priority is the establishment of **formal LLM governance** — including prompt-version tracking, fairness auditing, and privacy-preserving enrichment — to ensure accountability in regulated environments [15][16]. Extending this hybrid scoring framework beyond fraud detection to **anti-money-laundering (AML)** screening, **know-your-customer (KYC)** risk analysis, and **real-time transaction monitoring** may further expand its applicability across other compliance-critical domains.

**Conclusions**.

This paper presented a reproducible framework for AI-enhanced fraud detection in financial workflows, integrating machine learning (ML) and large language models (LLMs) under strict real-time and compliance constraints. The hybrid pipeline — covering ingestion, normalization, semantic enrichment, risk scoring, and auditable decision layers — demonstrates that explainability and latency can coexist without compromise. Assigning the LLM to an asynchronous enrichment path preserves deterministic response times while adding contextual intelligence and interpretability [5][6].

Empirical evaluation on real transaction data showed consistent gains in AUC-PR and Recall@FPR = 1 %, fewer false positives, and improved model stability compared with rule-based and classical ML baselines [7][11]. Human-readable rationales produced by the LLM bridge the gap between automated detection and auditability,

supporting transparent decision-making [6][12]. Operationally, the framework reduces manual-review workload and regulatory risk while maintaining customer trust.

Remaining challenges include label latency, data quality, and LLM inference cost. Future research should explore lightweight graph-based modules, online calibration for drift adaptation, and formal LLM governance to ensure fairness and accountability. Overall, the study delivers a trustworthy, explainable, and latency-aware AI architecture that advances practical, compliant antifraud systems for financial institutions.

**References:**

1. Jena, Sahil & Rajput, Divyansh & Pathak, Tatsat. (2025). Fraud Detection – A Hybrid Machine Learning Approach. International Journal of Scientific Research in Computer Science, Engineering and Information Technology. 11. 3463-3470. 10.32628/CSEIT25112825.

2. Chad, Felix. (2025). Explainable AI (XAI) in Financial Fraud Detection Systems.

3. SUMIT, SUMIT. (2025). Enhancing Card Fraud Detection Using Large Language Model (LLM). INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT. 09. 1-9. 10.55041/IJSREM50999.

4. Idowu, Marvel. (2025). Explainable AI for Financial Fraud Detection: Enhancing Transparency in Big Data-Driven Banking Systems.

5. Dohare, Anand & Deshpande, Uttam & Dahiya, Aman & Dabre, Kanchan & Srivastava, Kriti & Bhukya, Sreedhar & Sholapurapu, Prem Kumar. (2025). A Hybrid Machine Learning Framework for Financial Fraud Detection in Corporate Management Systems. EKSPLORIUM. 46. 139-154.

6. Faruk, Nayab & Tariq, Ahmad & Oladele, Sunday & Gok, Mooale. (2025). Explainable AI (XAI) for Fraud Detection: Building Trust and Transparency in AI-Driven Financial Security Systems.

7. Ajax, Raymond & kuforiji, john. (2025). AI for Financial Fraud Detection:

Real-time Anomaly Detection Systems.

8. Clement, Mateo. (2025). An Explainable AI Framework for Fraud Detection in Financial Ecosystems: Balancing Accuracy, Interpretability, and Compliance.

9. Srinivasa, Kalyan & Vangibhurathachhi, & Vangibhurathachhi, Srinivasa Kalyan. (2025). Machine Learning for Fraud Detection in Financial Transactions. 10.13140/RG.2.2.24887.23200.

10. Shanaa, Mohammad & Abdallah, Sherief. (2025). A Hybrid Anomaly Detection Framework Combining Supervised and Unsupervised Learning for Credit Card Fraud Detection. F1000Research. 14. 664. 10.12688/f1000research.166350.1.

11. Almalki, Fahad & Masud, Mehedi. (2025). Financial Fraud Detection Using Explainable AI and Stacking Ensemble Methods. 10.48550/arXiv.2505.10050.

12. Abill, Robert & Okunola, Abiodun & Abigail, Michack. (2025). Enhancing Fraud Detection through Hybrid AI Models: Combining Rule-Based Systems with Machine Learning.

Статья відправлена: 15.11.2025 г.