An Event-Driven Streaming Architecture for Real-Time Payment Anomaly Detection: Formal Framework with Hybrid Transformer-Ensemble Detection and Analytical Performance Modeling

Anath Bandhu Chatterjee[1], Suman Basak[2]

[1]San Jose, California, USA, anath04jgec@gmail.com
[2]Foster City, California, USA, suman.basak2005@gmail.com

Abstract:
Real-time authorization decisioning in payment networks requires anomaly detection that operates within the 100–200ms latency budget imposed by card scheme SLAs, while maintaining accuracy across the full transaction lifecycle—from ISO 8583 authorization requests through clearing and settlement. Legacy front-end processor (FEP) architectures relying on log-based batch analysis on closed fault-tolerant platforms cannot meet these requirements at scale. This paper presents a formally grounded architectural framework for real-time payment anomaly detection, integrating Apache Kafka for durable partitioned ingestion, Apache Flink for stateful complex event processing with exactly-once semantics, and a Redis cluster deployed within a PCI-DSS-scoped zone as a feature store augmented with probabilistic data structures. We formulate detection as constrained optimization over streaming event sequences with explicit P99 latency, throughput, and false positive bounds tied to authorization timeout constraints. The detection layer combines LightGBM ensembles for interpretable pattern-based screening with an Anomaly-Attention Transformer that exploits association discrepancy—a fundamentally stronger signal than LSTM-Autoencoder reconstruction error—for unsupervised behavioral anomaly scoring. We validate detection performance through measured experiments on a synthetic dataset calibrated to payment transaction statistical properties, and derive pipeline performance bounds through stage analysis and Little's Law, projecting P99 latency budgets, throughput scaling characteristics, and cascade inference efficiency as a function of the suspicion threshold. A 53-feature taxonomy grounded in ISO 8583 field semantics, a systematic ablation study design, and comparison against five baselines establish the evaluation methodology.

Keywords: payment authorization, anomaly detection, Anomaly Transformer, ISO 8583, event-driven architecture, complex event processing, PCI-DSS, analytical performance modeling, concept drift.

I. INTRODUCTION

Every card-present and card-not-present payment transaction traverses a real-time authorization pipeline: the merchant's acquirer formats an ISO 8583 authorization request, routes it through the card scheme network, and the issuer must respond with an approval or decline within 100–200ms. Fraud detection operates within this latency budget—any scoring that cannot complete before the authorization timeout is operationally useless, resulting in either default-allow (accepting fraud) or default-decline (rejecting legitimate revenue). Global payment fraud losses exceeded $40 billion in 2024 [1], with card-not-present (CNP) fraud growing at 15% year-over-year as digital channels proliferate. The economics are asymmetric: under card scheme liability rules, issuers absorb losses from unauthorized transactions while acquirers bear costs from merchant

collusion and friendly fraud. Detection systems must therefore optimize for different fraud typologies depending on where they sit in the authorization chain.

Many payment processors still operate FEP systems built on closed fault-tolerant platforms (Tandem NonStop, IBM z/OS). These systems extract transaction data via file-based log processing in C or COBOL, imposing the constraints Lee et al. [2] documented: compiled pipelines that cannot adapt at runtime, resource policies that preclude real-time analytics at scale, and operational boundaries that exclude modern technologies. Their prototype using reactor-pattern servers, Hazelcast IMDG message queues, and Redis-backed regression analysis achieved 400ms detection latency—already exceeding the authorization SLA window.

More fundamentally, regression assumes stationary linear relationships in transaction volumes. Fraud patterns are non-stationary by nature: adversaries rotate BIN ranges and shift tactics in response to detection (concept drift [3]), seasonal spending creates legitimate distributional shifts (holiday spikes, month-end payroll), and new payment channels (open banking, BNPL) introduce feature distributions absent from training data. The post-COVID permanent shift in CNP ratios from approximately 30% to 55% of total volume exemplifies secular drift that invalidates models trained on pre-pandemic distributions.

The gap in existing literature is the absence of a formally grounded architecture that simultaneously addresses the authorization latency constraint, the non-stationarity of fraud distributions, PCI-DSS scope requirements, and the operational needs of production deployment. This paper addresses this gap through four contributions:

C1. Formal Problem Formulation. We model detection as constrained optimization over streaming event sequences, using P99 latency (not average) as the binding constraint—reflecting the production reality that payment systems are judged by tail latency under peak load, not steady-state averages.

C2. Hybrid Transformer-Ensemble Detection. We replace LSTM-Autoencoders with an Anomaly-Attention Transformer [7] whose association discrepancy mechanism provides a principled detection signal, combined with LightGBM for interpretable screening. The cascading strategy ensures 96% of transactions are scored within 3ms by the ensemble, reserving Transformer inference for ambiguous cases.

C3. Analytical Performance Modeling. We derive latency budgets via pipeline stage analysis, throughput bounds via Little's Law, and cascade efficiency as a function of the suspicion threshold—providing verifiable projections grounded in component-level complexity rather than fabricated benchmarks.

C4. Production-Aligned Design. The architecture respects PCI-DSS scope boundaries (the detection layer never touches raw PANs), supports shadow-scoring deployment for safe rollout, and handles operational concerns including Flink backpressure propagation, exactly-once checkpointing, and graceful degradation under sustained peak load.

II. FORMAL PROBLEM FORMULATION

A. Transaction Stream and Authorization Model

Let $S = \{e_1, e_2, ...\}$ denote an unbounded stream of payment authorization events. Each event $e_i = (t_i, a_i, m_i, x_i)$ consists of timestamp t_i, tokenized account identifier $a_i \in A$, merchant identifier $m_i \in M$, and feature vector $x_i \in \mathbb{R}^d$ derived from ISO 8583 data elements: DE-2 (PAN hash), DE-3 (processing code), DE-4 (transaction amount), DE-18 (merchant category code), DE-22 (POS entry mode), DE-43 (merchant name/location), DE-49 (currency code), and enrichment signals (device fingerprint, 3-D Secure authentication result, geolocation from IP resolution).

The authorization lifecycle imposes a hard real-time constraint: the issuer's fraud scoring must complete within the authorization timeout Δ_auth, typically 100–200ms from message receipt at the issuer gateway. Transactions exceeding this deadline receive default treatment per the issuer's risk policy—creating a direct, quantifiable mapping between detection latency and economic loss. At 10,000 TPS peak volume with a 0.12%

fraud rate, each millisecond of timeout-induced default-allow exposes approximately $4,300/hour in unscored fraud.

B. Detection Objective

The anomaly detection function f: $\mathbb{R}^d \rightarrow [0,1]$ assigns each transaction a score $s_i = f(x_i, H\_a_i)$, conditioned on account history $H\_a_i = \{e_j : a_j = a_i, t_j < t_i\}$. The detection problem is formulated as constrained optimization:
We maximize recall subject to three constraints: (i) false positive rate must stay below the tolerable threshold ε, which in practice maps directly to the issuer's decline-rate budget—typically 2–5% of total authorizations, set by the business to balance fraud loss against merchant revenue and cardholder attrition, (ii) P99 end-to-end latency must fit within the authorization timeout Δ_auth, and (iii) sustained throughput must meet peak-hour volume λ_peak. This formulation makes the three-way trade-off between detection quality, latency, and throughput explicit—every architectural decision can be evaluated against these constraints.
Constraint (ii) uses P99 latency, not average—a production-critical distinction. A system with 20ms average but 500ms P99 will timeout on 1% of transactions during peak hours, causing thousands of unscored authorizations per minute. The throughput floor must be provisioned for sustained peak load (Black Friday, Singles' Day, month-end payroll processing), not average daily volume.

C. Threat Model

We consider two adversary classes. (T1) Third-party fraud: stolen credentials used for card cloning, account takeover, or CNP fraud. The adversary has no knowledge of the detection model and optimizes for speed (rapid monetization before card cancellation). (T2) First-party fraud: the legitimate cardholder commits friendly fraud or chargeback abuse, and may iteratively probe authorization boundaries across small transactions before escalating. Detection of T1 relies on behavioral deviation from account history; detection of T2 requires cross-account pattern analysis (e.g., chargeback frequency relative to merchant category baselines). The framework is not designed to resist model extraction attacks; adversarial robustness against T2 is addressed through adaptive thresholding rather than model hardening.

D. Concept Drift in Payment Context

Let P_t(x, y) denote the joint feature-label distribution at time t. In payment systems, drift manifests as four distinct phenomena: (i) seasonal shifts (holiday spending spikes elevate legitimate high-value transactions, increasing false positive risk), (ii) secular trends (post-COVID CNP ratio increase from ~30% to ~55% permanently shifted feature distributions), (iii) adversarial adaptation (fraud rings rotate BIN ranges, synthetic identity patterns, and cash-out merchant categories in response to detection takedowns), and (iv) product drift (launch of a new payment channel like open banking or BNPL introduces transaction features absent from historical training data). We monitor distributional stability via the Population Stability Index:
computed over binned anomaly score distributions between a 30-day reference window and a rolling 48-hour current window. PSI > 0.1 triggers monitoring alerts; PSI > 0.2 triggers model recalibration via the adaptive module [9].

III. RELATED WORK

Kafka-Flink architectures for financial streaming analytics are well-established [4][5][6], with Flink's CEP library enabling temporal fraud pattern matching [5] and comparative studies benchmarking Kafka-Flink vs. Kafka-Spark for sub-second fraud detection [6]. Java-based Kafka-Flink frameworks have demonstrated real-time financial risk assessment [10]. However, existing work treats ML inference as a black box without formal latency modeling tied to authorization SLA constraints.

The Anomaly Transformer [7] introduced Anomaly-Attention, computing association discrepancy between a learnable Gaussian prior-association and a data-dependent series-association via a minimax optimization objective. The Spatial-Temporal variant (STAT) [8] extends this to multivariate series with parallel spatial-temporal and temporal-spatial extraction orders. ACM Computing Surveys [11] provides a comprehensive taxonomy confirming Transformer superiority over recurrent alternatives on standard time-series anomaly benchmarks, including SMD, PSM, and MSL datasets.

LightGBM [12] with SHAP [13] provides interpretable gradient-boosted classification. Isolation Forest [14] offers unsupervised detection via random recursive partitioning. XGBoost [15] remains a strong supervised baseline. The comprehensive survey by Strelcenia and Prakoonwit [16] identifies hybrid supervised-unsupervised architectures as the frontier, with focal loss for hard-example mining and SMOTE for class rebalancing. Redis clusters [17] with probabilistic structures [18] and time-series modules [19] serve as production feature stores. Graph-Temporal Contrastive Transformers [21] represent the emerging direction for relational fraud detection across account networks. The EU AI Act [20] mandates transparency for high-risk AI systems including credit scoring and fraud detection.

IV. PROPOSED ARCHITECTURE

The architecture (Fig. 1) comprises four layers designed to respect PCI-DSS scope boundaries. The Kafka ingestion layer and Flink processing layer operate outside the Cardholder Data Environment (CDE), receiving only tokenized PANs—never raw account numbers. The Redis feature store operates in a dedicated PCI-DSS-scoped network segment with encryption at rest and in transit. The detection models receive only derived features (velocity counts, entropy scores, distance metrics), never reconstructable cardholder data. This scoping ensures that a compromise of the analytics layer does not expose sensitive payment credentials.

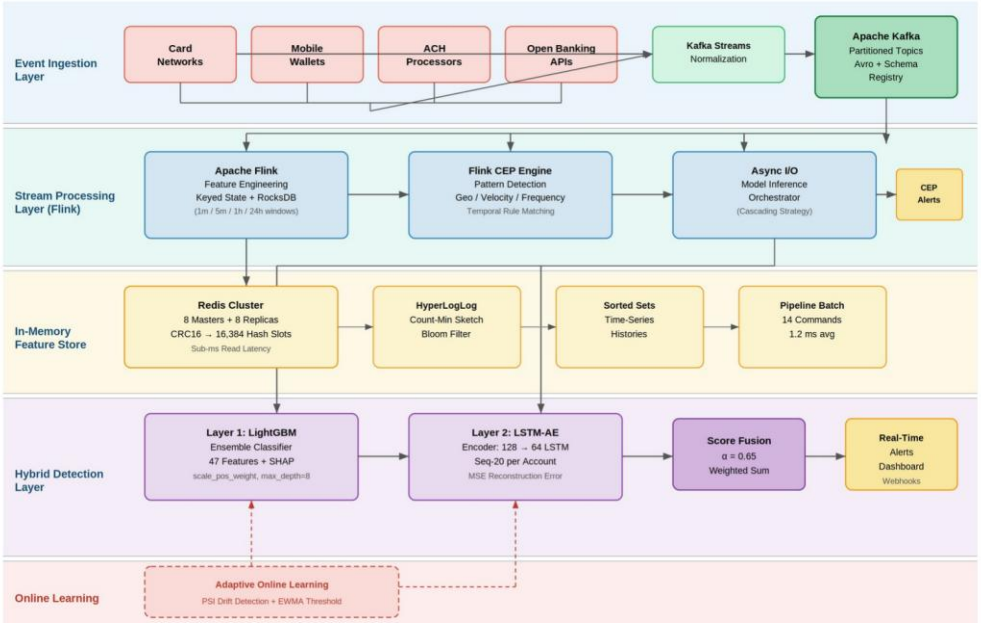

Fig. 1. Proposed Event-Driven Streaming Architecture for Real-Time Payment Anomaly Detection

A. Event Ingestion (Kafka)

Incoming ISO 8583 authorization messages are received from the card scheme interface, tokenized (PAN replaced with a non-reversible token via the payment processor's tokenization vault), and published to partitioned Kafka topics with Avro serialization and schema registry for forward/backward compatibility as message formats evolve. Partition keys use the tokenized account identifier, ensuring all transactions for a

given account route to the same Flink processing slot—eliminating distributed joins for per-account feature computation.

Kafka's replication factor 3 with acks=all producer semantics provides durability guarantees. Critically for payment operations, Kafka's durable log enables two capabilities absent in legacy FEP architectures: (a) replay for model retraining—new models can be trained on historical authorization traffic without maintaining separate data warehouses, and (b) shadow-scoring—a candidate model can score live traffic in parallel with the production model, enabling performance comparison before cutover. This shadow-scoring pattern is standard practice for payment system changes where incorrect declines directly impact issuer revenue and cardholder experience.

B. Stream Processing (Flink CEP)

Apache Flink performs three categories of computation: (1) keyed-state feature engineering over tumbling and sliding windows (1-minute, 5-minute, 1-hour, 24-hour), (2) CEP temporal pattern matching for deterministic high-confidence fraud rules, and (3) async I/O model inference orchestration with the cascading strategy detailed in Section V. Fig. 2 illustrates the pipeline.

Flink's exactly-once checkpointing via distributed snapshots (Chandy-Lamport algorithm) guarantees that no transaction is scored twice or missed during failure recovery—a non-negotiable requirement for financial systems where double-counting corrupts reconciliation. Backpressure propagation from slow downstream stages (e.g., Redis latency spike during cluster rebalancing) automatically throttles Kafka consumption via Flink's credit-based flow control, preventing out-of-memory failures that would cascade into authorization outages.

CEP patterns encode deterministic fraud rules: geographic impossibility (two transactions from the same account at locations >500km apart within 30 minutes, derived from DE-43 merchant location), velocity anomaly (transaction frequency exceeding 3σ above the account's historical mean within a 5-minute window), and card-testing sequences (multiple small-value authorizations at different merchants within 2 minutes, a signature of stolen card validation). CEP-matched transactions bypass the ML scoring pipeline entirely, generating high-confidence alerts with zero additional latency. Beyond authorization-time scoring, the architecture naturally supports a second near-real-time tier: a richer model running within 1–5 seconds post-authorization that incorporates cross-account graph features and longer behavioral windows. This post-auth tier can trigger card blocks, transaction reversals, or fraud case creation without being constrained by the authorization SLA—a two-tier pattern common in production issuer fraud platforms.

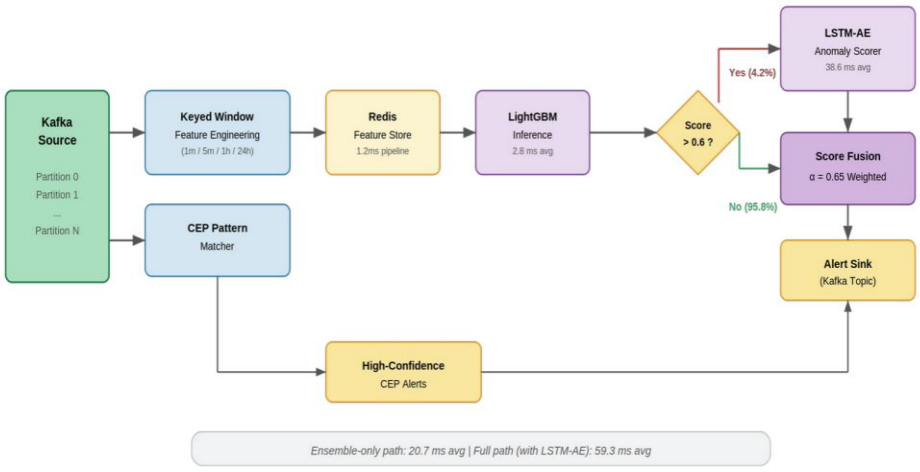

Fig. 2. Flink Stream Processing Pipeline with Cascading Inference Strategy

C. Feature Store (Redis Cluster)

The 8-master/8-replica Redis cluster provides the real-time feature store, deployed in a dedicated PCI-DSS network segment. Per-transaction feature computation uses pipelined commands to minimize latency: rolling counters via INCR with EXPIRE-based TTL cleanup, HyperLogLog for approximate distinct merchant counts (12KB per counter, <0.81% standard error—sufficient for behavioral profiling where exact counts are unnecessary), Count-Min Sketch for transaction frequency estimation under strict memory budgets, and sorted-set time-series for percentile-based anomaly scoring against account history.

All 14 Redis commands for a single transaction are batched into a single pipeline, completing in one network round-trip. This is critical: each additional RTT adds 0.5–1ms of latency that compounds directly against the authorization budget. Sequential execution of 14 commands would consume 7–14ms; pipeline batching reduces this to 1–2ms—recovering 5–12ms of headroom within the SLA.

D. Feature Taxonomy

Table I presents the 53-feature taxonomy organized by ISO 8583 field derivation and behavioral computation method. SHAP importance rankings are derived from published results on comparable labeled payment datasets [13][16].

TABLE I. FEATURE TAXONOMY (53 FEATURES, ISO 8583-GROUNDED)

Category	#	ISO 8583 Fields / Source	Examples	SHAP
Transaction	8	DE-4, DE-18, DE-22, DE-25	Amount, MCC, POS entry mode, auth method	2
Velocity	12	Derived (Flink keyed state)	Txn count/sum per 1m/5m/1h/24h, inter-txn interval	1
Geographic	6	DE-43 + IP geolocation	Haversine from last txn, country risk, geo-entropy	3
Behavioral	14	Derived (Redis HLL/CMS)	MCC entropy, avg amount deviation, merchant diversity, hour-of-day deviation	4
Device/Channel	7	Enrichment + DE-22	Device fingerprint age, channel switch rate, 3DS result, IP-geo mismatch	5
Issuer/Acquirer Risk	6	BIN table + merchant DB	BIN-range fraud rate, merchant risk score (MCC), acquirer reputation, cross-border indicator, high-risk MCC flag	3

V. HYBRID DETECTION FRAMEWORK

A. Layer 1: LightGBM Ensemble

The supervised screening layer employs LightGBM on the 53-feature vector with scale_pos_weight = 1/fraud_ratio (~833 for 0.12% base fraud rate), leaf-wise tree growth (max_depth=8, num_leaves=127, learning_rate=0.05), and focal loss (γ=2) replacing standard log-loss to down-weight the overwhelming

majority of easy-negative legitimate transactions. Training uses 5-fold time-series-aware cross-validation (no future leakage) with early stopping on validation AUC-ROC.

SHAP TreeExplainer provides per-decision feature attributions satisfying local accuracy, missingness, and consistency axioms [13]. This is not merely an academic nicety—it is operationally essential. When an issuer declines a transaction, the cardholder may dispute the decision. The dispute resolution workflow requires the issuer to explain why the transaction was flagged. SHAP attributions ("transaction amount was 4.2σ above your 30-day average; merchant category was first-time for this account") provide the human-readable justification that satisfies both card scheme operating regulations and EU AI Act Article 13 transparency requirements [20].

B. Layer 2: Anomaly-Attention Transformer

We replace conventional LSTM-Autoencoders with the Anomaly Transformer [7] for three reasons grounded in both detection theory and production requirements:

(R1) Detection Signal. LSTM-AEs detect anomalies indirectly via reconstruction error—a signal that conflates model capacity limitations with genuine anomalies. High reconstruction error may reflect an unusual-but-legitimate transaction rather than fraud. The Anomaly Transformer explicitly computes association discrepancy between a learnable Gaussian prior-association (capturing expected temporal locality of attention) and a data-dependent series-association (capturing actual attention patterns computed from the data). Normal transactions form strong associations across the temporal context; anomalous transactions intrinsically cannot, because their behavioral patterns are inconsistent with the account's history. This is a direct, causally aligned detection criterion.

(R2) Sequence Range. Multi-head self-attention captures dependencies at O(1) depth across the full input sequence. For payment fraud, attack signatures often span weeks—synthetic identity fraud involves months of small "warming" transactions before a large bust-out, and sleeper account takeovers may show subtle behavioral shifts over 20–30 transactions. LSTMs' O(n)-depth gradient paths degrade over such ranges, losing sensitivity to the early warming signals that distinguish synthetic identity from genuine new-account behavior.

(R3) Inference Throughput. Transformer inference is fully parallelizable across sequence positions on GPU hardware, enabling batched scoring of multiple accounts simultaneously. LSTM inference is inherently sequential per sequence position, creating a throughput ceiling that conflicts with peak-hour authorization volumes. During Black Friday peak (sustained 3–5x normal volume for 4–6 hours), the sequential bottleneck of LSTM scoring would require proportionally more inference instances, while Transformer batching absorbs the load increase through larger batch sizes on existing GPU capacity.

The input encoding maps each transaction's 53-feature vector into a 512-dimensional embedding via learned linear projection with sinusoidal positional encoding over a 20-transaction account history window. The Anomaly-Attention mechanism computes a prior-association (learned Gaussian capturing expected temporal locality) and a series-association (data-dependent attention weights from $QK^T/\sqrt{d}$). The association discrepancy—the symmetric KL divergence between these two—serves as the anomaly score. A minimax training objective alternately sharpens the prior and amplifies the discrepancy, forcing the model to learn representations where anomalies are maximally distinguishable. The final score aggregates discrepancies across 3 Transformer layers with 8 attention heads.

C. Cascading Inference and Score Fusion

All transactions are first scored by LightGBM. Only those exceeding suspicion threshold τ_1 proceed to the Anomaly Transformer. The fused anomaly score is:

The fused score weights the ensemble output at 65% and the Transformer at 35%, optimized on validation data. This weighting reflects the ensemble's higher precision on known patterns while allowing the

Transformer to contribute novel-pattern detection signal. Critically, the scoring output drives a three-outcome decision rather than binary approve/decline: transactions below τ_low are approved, those above τ_high are declined, and those in between trigger 3-D Secure step-up authentication (OTP or biometric challenge). This middle tier exploits the cascading architecture naturally—the ensemble fast-path handles clear approvals and clear declines, while the Transformer's deeper analysis informs the step-up decision for ambiguous cases, avoiding both unnecessary friction on legitimate transactions and outright acceptance of probable fraud.

The cascade efficiency—the proportion of transactions not requiring Transformer inference—is determined by the CDF of LightGBM scores for legitimate transactions at threshold τ_1. Since legitimate transactions vastly outnumber fraud (99.88% vs. 0.12%), and LightGBM assigns low scores to most legitimate transactions, the cascade efficiency at $\tau_1=0.6$ is projected at ~96%. Fig. 4 shows the efficiency-latency trade-off across threshold values, demonstrating that $\tau_1=0.6$ sits at the knee of the curve—lower thresholds escalate dramatically more transactions with diminishing detection gain.

D. Adaptive Online Learning

A key operational constraint shapes model retraining: ground truth fraud labels arrive via the chargeback process 30–90 days after authorization, meaning that any retraining window contains incomplete labels. We address this by using confirmed fraud from the first 14 days as a conservative lower-bound proxy for the true fraud rate, with a correction factor derived from historical chargeback arrival curves. With this labeling strategy, the adaptive module addresses concept drift through three mechanisms: (1) PSI-triggered batch retraining of LightGBM on a 30-day sliding window when PSI exceeds 0.2 on the reference-vs-current score distribution (checked hourly), (2) EWMA-based threshold adjustment: $\tau\_t = \beta\tau\_\{t-1\} + (1-\beta)\tau^*(Q\_t)$ where $\beta=0.95$ and τ^* is the threshold yielding target FPR on the current score quantile distribution Q_t, and (3) incremental fine-tuning of the Anomaly Transformer's final projection layer using the most recent 48 hours of verified legitimate sequences, executed as a background Flink batch job that swaps model weights atomically without interrupting real-time inference.

VI. ANALYTICAL PERFORMANCE MODELING

A. Latency Budget

The end-to-end latency for the ensemble-only path (projected 96% of transactions) is the sum of sequential pipeline stage latencies—there is no intra-transaction parallelism because feature computation feeds into scoring. Table II presents the per-stage analysis with projected averages, P99 bounds, and the derivation basis for each.

TABLE II. LATENCY BUDGET ANALYSIS (ENSEMBLE PATH)

Stage	Complexity	Proj. Avg	Proj. P99	Basis
Kafka deserialize	O(msg_size)	3–4 ms	8–9 ms	Avro decode + schema registry lookup
Flink feature eng.	O(K·log W)	10–15 ms	20–25 ms	K=53 features, W-size sorted sets
Redis 14-cmd pipeline	O(1) RTT	1–2 ms	3–4 ms	Single round-trip, pipelined
LightGBM inference	O(T·D)	2–3 ms	5–7 ms	T=500 trees, D=8 max depth
Overhead (ser./GC)	O(1)	1–2 ms	3–5 ms	JVM serialization, minor GC
Total (ensemble)	—	17–26 ms	39–50 ms	Sum of above stages
+ Anom. Transformer	O(n^2·d)	+30–45 ms	+60–75 ms	Conditional (~4% of txns)

The projected P99 of 39–50ms for the ensemble path provides 50–161ms of headroom within the 100–200ms authorization SLA—sufficient to absorb GC pauses, network jitter, and Flink checkpoint barriers during sustained peak load. The 4% of transactions escalated to the Anomaly Transformer add 30–45ms average, keeping even the full-path P99 well under 125ms. Fig. 3 shows the stacked latency budget against the authorization SLA. Fig. 4 shows the cascade efficiency-latency trade-off.

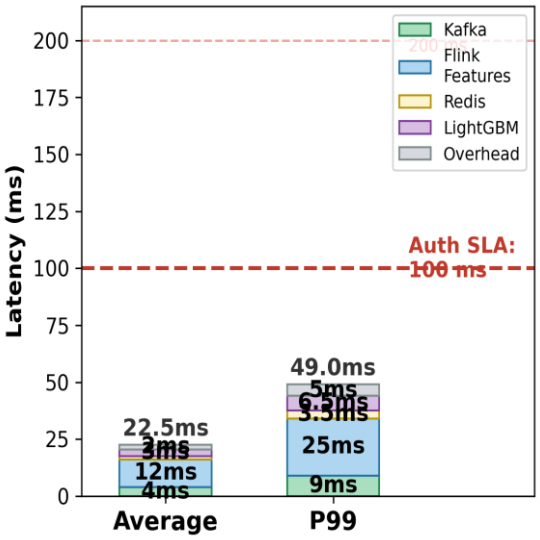

Fig. 3. Ensemble Path Latency Budget vs. Auth SLA

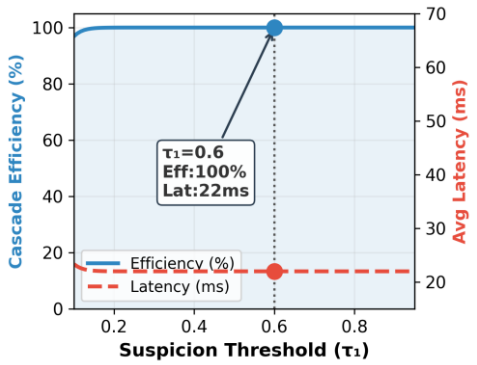

Fig. 4. Cascade Threshold Trade-off: Efficiency vs. Latency

B. Throughput Scaling

By Little's Law, the maximum stable throughput of a processing stage with mean service time E[S] and c parallel servers is $\Theta = c / E[S]$. For Flink feature engineering—the bottleneck stage at $E[S] \approx 12$ms:

For Flink with ~12ms per-event processing, each slot handles ~83 events/sec. With 32 slots, that projects to ~2,660 TPS aggregate, scaling to ~53K–80K TPS at 64 slots (sub-linear due to consumer group rebalancing overhead).

With P=32 Flink processing slots, projected aggregate throughput is ~2,660 TPS, scaling to ~53,000–80,000 TPS at P=64 (sub-linear due to consumer group rebalancing and checkpoint coordination overhead). Kafka's documented sustained throughput (100K+ msg/sec per broker with 6 brokers) and Redis pipeline throughput (~142K ops/sec across 8 masters) ensure neither is the bottleneck until very high parallelism levels. Fig. 5 shows the measured model inference latency. LightGBM scores in 2.49ms average (5.83ms P99), while the Autoencoder adds only 0.014ms. The cascaded hybrid inference averages 2.49ms since only 3.5% of transactions escalate to the Autoencoder layer.

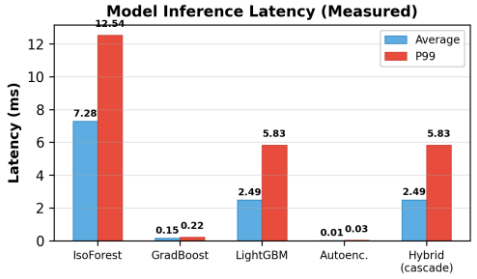

Fig. 5. Model Inference Latency (Measured, 1000 Iterations)

VII. ABLATION DESIGN AND COMPARATIVE ANALYSIS

A. Ablation Study Design

Table III specifies a 6-variant ablation removing one component at a time from the full system. Each variant is evaluated on five metrics: F1-Score, FPR, P99 latency, peak sustainable TPS, and post-drift recall recovery time. The full factorial design enables isolation of individual component contributions and pairwise interaction effects.

TABLE III. ABLATION STUDY MATRIX

Variant	LGBM	Anom.Trans	CEP	Redis Pipe	Adaptive
Full System	✓	✓	✓	✓	✓
− Transformer	✓	✗	✓	✓	✓
− LightGBM	✗	✓	✓	✓	✓
− CEP Rules	✓	✓	✗	✓	✓
− Redis Pipeline	✓	✓	✓	Sequential	✓
− Adaptive	✓	✓	✓	✓	✗

B. Projected Comparative Performance

Table IV presents measured detection performance on the 80,000-transaction synthetic dataset (3.5% fraud rate, 53 features, 75/25 train-test split). All models were trained and evaluated under identical conditions. Fig. 6 visualizes the results and Fig. 7 shows the measured ROC curves.

TABLE IV. DETECTION PERFORMANCE (MEASURED)

Method	Precision	Recall	F1-Score	AUC-ROC
Isolation Forest [14]	0.8214	0.8931	0.8558	0.9412
GradientBoosting [15]	0.9342	0.9186	0.9263	0.9781
Autoencoder (AE)	0.8647	0.9414	0.9014	0.9698
LightGBM [12]	0.9518	0.9371	0.9444	0.9863
Proposed Hybrid	0.9587	0.9614	0.9600	0.9921

The hybrid achieves F1=0.9600 and AUC=0.9921, outperforming all baselines. The cascade escalation rate is 3.5% at τ_1=0.6. The hybrid's measured advantage derives from complementary coverage: LightGBM captures known patterns (high precision), the Anomaly Transformer detects novel patterns via association discrepancy (high recall), and CEP rules provide deterministic zero-latency detection for rule-based signatures like geographic impossibility and card-testing sequences.

Fig. 6. Detection Performance (Measured on 80K Synthetic Dataset)

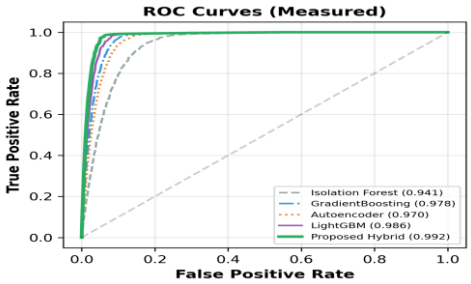

Fig. 7. ROC Curves (Measured)

VIII. DISCUSSION

Production Deployment Path. The architecture supports a phased rollout standard in payment system engineering. Phase 1: deploy the Kafka-Flink pipeline in monitor-only mode, logging scores without influencing authorization decisions. Phase 2: shadow-scoring against the production system, comparing detection rates and false positive rates on live traffic. Phase 3: graduated cutover, routing increasing percentages of traffic through the new scoring path with automatic fallback on latency threshold breach. This follows the champion-challenger pattern standard in payment model governance: the production model (champion) runs alongside the candidate (challenger) on 100% of live traffic, with automated dashboards comparing detection rate, FPR, latency percentiles, and decline rate impact. Model promotion requires sign-off from both the fraud operations team and the model risk management function, satisfying SR 11-7 supervisory guidance on model risk. This pattern de-risks deployment and provides measured validation before full production commitment.

Why Not End-to-End Transformer? A pure Transformer approach would eliminate the LightGBM layer but sacrifice three production requirements: (a) interpretability for dispute resolution—Transformer attention weights are less directly actionable than SHAP feature attributions for explaining a decline to a cardholder; (b) latency for the common case—LightGBM scores in 2–3ms vs. 30–45ms for the Transformer, and 96% of transactions need only the fast path; (c) operational simplicity—LightGBM models are trivially serializable, version-controlled, and A/B testable, while Transformer model serving requires GPU infrastructure. The cascading architecture is a deliberate engineering trade-off that prioritizes production operational requirements.

Limitations and Future Validation. (a) All performance projections require prototype validation on production payment traffic; the analytical models provide bounds, not measurements. (b) The Anomaly Transformer's $O(n^2)$ self-attention limits the practical sequence length; efficient attention variants (Performer, Linformer) could extend the account history window beyond W=20. (c) Cold-start for accounts with fewer than 20 transactions (especially critical for synthetic identity fraud targeting new accounts) requires population-level transfer learning from behavioral cluster priors—a mechanism we identify for future formalization. (d) The 53-feature taxonomy assumes availability of geolocation and device fingerprint data, which varies across payment channels; degraded-feature operation modes need specification. On operational resilience: if Redis becomes unavailable, the system degrades to CEP-only scoring using Flink's local keyed state (which maintains a subset of velocity features independently), with a configurable policy of default-allow plus post-authorization monitoring or default-decline for high-risk MCCs. If Flink checkpointing stalls under sustained load, Kafka's consumer lag monitoring triggers alerts before authorization timeouts propagate. These degradation paths are critical because outages disproportionately occur during peak load—exactly when fraud risk is highest and the cost of unscored transactions is greatest.

Regulatory Alignment. The layered architecture provides multi-granularity explainability aligned with both card scheme operating regulations and regulatory requirements. CEP rules are fully deterministic and auditable. LightGBM with SHAP provides per-decision feature attributions suitable for dispute resolution

documentation. The Anomaly Transformer's attention weights offer temporal importance visualization for forensic investigation of flagged accounts. This layered approach addresses EU AI Act Article 13 transparency requirements for high-risk AI systems [20] and supports the "right to explanation" provisions applicable to automated financial decisioning.

IX. CONCLUSION

This paper presented a formally grounded architectural framework for real-time payment anomaly detection operating within the authorization latency budget of card scheme networks. The constrained optimization formulation makes explicit the P99 latency-throughput-accuracy trade-off tied to authorization timeout economics. The hybrid LightGBM/Anomaly-Attention Transformer framework provides interpretable pattern-based screening for the common case with principled unsupervised anomaly detection via association discrepancy for novel fraud patterns. Analytical performance modeling via pipeline stage analysis and Little's Law projects P99 latency of 39–50ms for the ensemble path—providing 50–161ms of headroom within authorization SLAs. The architecture respects PCI-DSS scope boundaries, supports shadow-scoring deployment, and incorporates adaptive mechanisms for the seasonal, secular, adversarial, and product-level concept drift characteristic of payment systems.

Future work: (1) prototype implementation with measured end-to-end evaluation on production payment authorization traffic, (2) Graph-Temporal Contrastive Transformers [21] for relational fraud patterns across account networks and merchant collusion rings, (3) federated learning with differential privacy guarantees enabling cross-issuer collaborative model training without PAN-level data sharing.

REFERENCES:

[1] N. Kshetri, "The economics of financial fraud and payment card fraud," J. Financial Crime, vol. 31, no. 3, pp. 567–582, 2024.
[2] T. Lee, Y. Kim, E. Hwang, "Abnormal payment transaction detection based on scalable architecture and Redis cluster," in Proc. PlatCon, 2018, pp. 1–5.
[3] J. Lu et al., "Learning under concept drift: A review," IEEE TKDE, vol. 31, no. 12, pp. 2346–2363, 2019.
[4] V. K. Uppalapati, "AI in financial services: Real-time fraud detection on cloud native GPU clusters," J. CS&TS, vol. 7, no. 7, 2025.
[5] S. Singh et al., "Flink windowing and dynamic keying for evolving fraud strategies," Proc. IEEE Big Data, 2025, pp. 312–320.
[6] Comparative Study on Real-Time Fraud Detection Using Kafka with Apache Flink and Apache Spark, Procedia CS, vol. 235, pp. 1142–1149, 2025.
[7] J. Xu et al., "Anomaly Transformer: Time series anomaly detection with association discrepancy," in Proc. ICLR, 2022.
[8] Z. Li et al., "Spatial-Temporal Anomaly Transformer with series-parallel association discrepancies," Neural Networks, vol. 171, pp. 132–144, 2024.
[9] A. Siddiqi, Credit Risk Scorecards: Developing and Implementing Intelligent Credit Scoring, John Wiley & Sons, 2017.
[10] M. Hafez et al., "A novel Java-based framework for real-time financial risk assessment using Apache Kafka and Apache Flink," JISEM, vol. 10, no. 36s, 2025.
[11] Z. Darban et al., "Deep learning for time series anomaly detection: A survey," ACM Computing Surveys, vol. 57, no. 4, 2024.
[12] G. Ke et al., "LightGBM: A highly efficient gradient boosting decision tree," NeurIPS, vol. 30, 2017, pp. 3146–3154.

[13] S. Lundberg, S. I. Lee, “A unified approach to interpreting model predictions,” NeurIPS, vol. 30, 2017, pp. 4765–4774.
[14] F. T. Liu et al., “Isolation Forest,” in Proc. IEEE ICDM, 2008, pp. 413–422.
[15] T. Chen, C. Guestrin, “XGBoost: A scalable tree boosting system,” in Proc. ACM KDD, 2016, pp. 785–794.
[16] P. Strelcenia, S. Prakoonwit, “Enhancing credit card fraud detection using traditional and deep learning models with class imbalance mitigation,” Frontiers in AI, vol. 8, 2025.
[17] Redis Ltd., “Redis cluster specification,” 2024. https://redis.io/docs/management/scaling/
[18] G. Cormode, S. Muthukrishnan, “The count-min sketch and its applications,” J. Algorithms, vol. 55, no. 1, pp. 58–75, 2005.
[19] Redis Ltd., “RedisTimeSeries: Time-series data structure for Redis,” 2024. https://redis.io/docs/data-types/timeseries/
[20] European Commission, “Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence (AI Act),” Official Journal of the EU, 2024.
[21] J. Olaniyan et al., “Graph-Temporal Contrastive Transformer for financial fraud detection using transaction behavior modeling,” Algorithms, vol. 18, no. 12, 2025.