Journal of Business and Management Studies (JBMS)
ISSN: 2709-0876
DOI: 10.32996/jbms
Journal Homepage: www.al-kindipublisher.com/index.php/jbms

JBMS
AL-KINDI CENTER FOR RESEARCH AND DEVELOPMENT

| RESEARCH ARTICLE

Real-Time Hybrid Optimization Models for Edge-Based Financial Risk Assessment: Integrating Deep Learning with Adaptive Regression for Low-Latency Decision Making

Md Parvez Ahmed[1], Sanjida Akter Tisha[2] and Md Murshid Reja Sweet[3]
[1]*Master of Science in Information Technology, Washington University of Science and Technology, USA*
[2]*Master of Science in Information Technology, Washington University of Science and Technology, USA*
[3]*Department of Management Science and Quantitative Methods, Gannon University, USA*
Corresponding Author: Md Parvez Ahmed, **Email:** sweet006@gannon.edu

| ABSTRACT
Financial institutions in the United States face mounting pressure to detect fraud and evaluate transaction risk in real time across highly distributed payment infrastructures, including mobile banking, point-of-sale devices, and ATM networks operating on resource-constrained hardware. Deep learning models deliver strong predictive accuracy for fraud detection but often exceed strict latency budgets when executed at the edge. Conversely, lightweight regression systems provide rapid decision-making but sacrifice accuracy under nonlinear transaction behaviors common in U.S. financial environments. This study develops a real-time hybrid optimization framework that dynamically integrates deep neural inference with adaptive regression, guided by an online controller that monitors latency, compute utilization, and confidence thresholds per transaction. Using a U.S. credit card fraud dataset structured as a streaming financial workload, we benchmark hybrid performance against standalone deep learning and regression baselines under simulated edge CPU and memory constraints. Experiments show that hybrid routing reduces inference latency by up to 55 percent compared to deep learning alone, while preserving high recall on fraudulent cases and improving transaction-level risk detection without overwhelming edge hardware. A latency-accuracy Pareto analysis highlights the system's ability to maintain regulatory-aligned response times without destabilizing detection performance, demonstrating practical readiness for deployment in payment terminals and digital banking infrastructure. These findings suggest that real-time model optimization can significantly enhance operational compliance, fraud resilience, and customer experience across U.S. financial systems, which are increasingly dependent on edge-based decision intelligence.

| KEYWORDS
Edge Computing, Financial Risk, Hybrid Modeling, Real-Time Optimization, Latency-Aware AI, Adaptive Regression

| ARTICLE INFORMATION
ACCEPTED: 15 October 2024 **PUBLISHED:** 03 November 2025 **DOI:** 10.32996/jbms.2025.7.7.5

1. Introduction

1.1 Background and Motivation

Financial fraud remains one of the most persistent operational threats within the United States banking and payment infrastructures, where billions of dollars are lost annually due to real-time transactional exploitation. Traditional fraud detection pipelines frequently rely on centralized servers that process large batches of transactions, but this approach introduces delays that can undermine fraud interception. As financial ecosystems shift toward instant digital payments, contactless transactions, and a growing dependence on mobile devices and distributed hardware, the latency vulnerabilities in centralized systems become increasingly unacceptable. Deep learning has demonstrated strong effectiveness in modeling highly nonlinear financial transaction patterns, especially when risk signals are hidden in high-dimensional behavior. Clements et al. (2020) highlight that sequential deep learning applied to tabular financial transaction data can significantly enhance credit risk monitoring accuracy by capturing evolving risk signatures in real time [4]. Recurrent neural networks have shown additional promise in the early

detection of broader financial system instabilities, underscoring the importance of temporal modeling in emerging threats. Tölö (2020) argues that recurrent networks enable the predictive identification of systemic crisis triggers, improving resilience across financial networks that depend on rapid decision cycles [23].

Neural approaches in fraud detection have a long history. Researchers were exploring these ideas decades ago. Ghosh and Reilly (1994) showed that neural networks trained on streams of payment activity could pick up signals that rigid rule systems failed to notice. Their work helped move fraud detection toward models that learn from behavior rather than relying on static thresholds [10]. That early insight still holds value today. As neural models grew larger and demanded more computational power, teams in finance pushed for higher accuracy. The pressure to spot complex, hidden fraud patterns encouraged deeper and heavier architectures. This pursuit created an uncomfortable reality for real-world deployment. These models placed a heavy load on the hardware sitting inside point-of-sale devices, ATM security modules, or card terminals. Decisions need to arrive in a fraction of a second in those settings. Anything slower risks blocking legitimate customers in awkward situations or letting suspicious transactions slip through because the model cannot finish its work in time. Retail banking in the United States adds even more urgency. Real-time payment rails like FedNow aim for near-instant authorization to keep commerce moving and keep customers confident that their bank is reliable. That speed expectation leaves zero room for hesitation. Every millisecond carries a dollar value and a reputation cost. This has increased interest in placing stronger intelligence directly at the transaction source. Running inference on the edge offers both speed and a tighter privacy boundary. The main hurdle lies in front of us. Financial institutions need models that can learn nonlinear fraud behavior while still fitting inside the narrow compute budgets of edge hardware. Teams designing these systems face a difficult balance: smarter pattern recognition on one side, immediate decision-making on the other. Until that gap closes, fraud systems will continue to face the constant risk of either missing key warning signs or reacting too slowly to stop them when it matters most.

1.2 Importance of This Research

Financial institutions in the United States walk a tightrope. They need to protect customers and the integrity of payments through strong risk assessments while keeping every interaction smooth and instant. This has grown harder as banking moves into pockets, checkout terminals, cars, and connected devices. These systems sit on distributed hardware that cannot always rely on the cloud as the primary brain, especially when reliability is the first priority. Feedzai (2023) points out that attackers take advantage of any millisecond of hesitation in fraud detection systems, since a reaction that trails behind a transaction creates openings for loss [8]. Fraud tactics change quickly, and a delayed response invites both financial damage and reputational hits. Edge computing is showing up as a way to avoid those dangerous pauses. Running analytics close to where a transaction begins trims network delays and keeps services functioning even when connectivity wobbles. Research also shows that placing financial analytics on mobile edge devices can improve how organizations detect and respond to risk. Chunhua and Lijun (2022) argue that this setup gives companies a stronger chance to catch weaknesses than systems that depend on a distant data center [3]. The shift to the edge creates its own hurdles. Deep learning models need to run reliably on compact devices with limited compute. Weak performance in those conditions can lead to missed fraud attempts or flawed compliance reports.

Regulators add weight to the challenge. US financial rules expect fast intervention when a transaction looks suspicious, and they ask institutions to explain the reasoning behind those alerts. Slow systems raise the chance of non-compliance. Fast systems that guess poorly increase exposure to fraud. Institutions need a thoughtful balance of speed, accuracy, clarity, and stability. The opportunity is in hybrid modeling that reacts to the environment in real time and still produces decisions that stand up to scrutiny. Strengthening fraud detection at the same edge locations that drive modern consumer payments can give institutions a path forward that improves protection without expanding attack surfaces or infrastructure spend.

1.3 Research Objectives and Contributions

This work looks at a simple question that turns out to be quite complicated in practice: how do you keep financial risk modeling both smart and fast when the real world is messy and the hardware sitting at the edge is not always powerful? The research builds a decision system that blends deep learning with models that stay lightweight and transparent. The goal is to let each transaction receive the best possible screening based on what the device can handle in that moment, without weakening fraud protection or shaking trust in the system. The idea centers on a pipeline that pays attention to its own confidence and resource use during inference. Every payment request is judged not only by the features in the data but also by how much time and computing resources are available. The model can choose a path that finishes in milliseconds. That kind of responsiveness matters because a small delay can open the door to losses. The work relies on data patterns that mirror real behavior in the U.S. financial system. This ensures that results do not stop at theory. It also introduces a clear way to measure the relationship between accuracy, latency, and regulatory expectations, which gives banking teams something concrete to weigh when deciding how far to push analytics out to the edge. The vision is straightforward. Build systems that protect people in the exact moment a transaction happens. Keep them secure, fast, interpretable, and ready for the pressure of real financial environments. This research takes a step in that direction by showing that adaptability, not fixed rules, can define the future of fraud detection.

2. Literature Review

2.1 Real-Time Financial Risk Modeling

Real-time fraud detection in the United States financial systems depends on fast ingestion and analysis of streaming transaction data, where risk must be assessed before authorization is completed. Industry reports indicate growing reliance on machine learning pipelines designed to monitor anomalies instantaneously within high-velocity payment streams. Alloy (2023) notes that financial institutions depend on continuous model scoring to mitigate fraud exposure during mobile and card-present transactions, where every transaction is scrutinized in milliseconds while ensuring minimal friction for legitimate users [1]. Tinybird (2023) further explains that scalable fraud detection now requires architectures capable of handling rapid state changes, complex event relationships, and adaptive rules triggered by transactional context, emphasizing that slow cloud-routed inference increases exposure as fraudulent events propagate through networks faster than legacy systems can analyze [22]. As models improve in predictive precision, adversaries evolve their tactics, increasing the need for systems that update risk assessments in real time rather than post-analysis.

This pace exposes a deeper issue. Many banks still depend heavily on centralized cloud computing to process transactions. That design can become unreliable when the network hesitates or when servers get overloaded. Even small delays create windows where criminals move money through before alarms go off. To avoid that, more organizations are placing pieces of the decision logic closer to where payments originate. A phone, a point-of-sale terminal, or an ATM begins to share responsibility for the call. That shift introduces new challenges because devices in the field do not behave the same way. Some have strong processors. Some barely handle their own baseline tasks. Connectivity varies minute by minute. If risk models demand too much computation, the whole system slows, and payment networks cannot tolerate slow. Research in this area focuses on making models accurate, fast, and resilient under uneven conditions. The goal is consistency at a national scale, even when millions of devices submit decisions at once. Every millisecond carries weight because a small delay can turn risk into loss. The work here is not only about clever modeling techniques. It is also about engineering defenses that hold up under real-world pressure, where uncertainty is the default state and financial harm can escalate before anyone notices.

2.2 Deep Learning in Financial Risk

Deep learning has become a primary driver of modern financial risk analytics because it captures nonlinear dependencies and hidden temporal fraud patterns that generalized linear models fail to represent effectively. Manzo and Qiao (2020) assert that neural architectures learn latent credit behavior signals across extended feature sets and temporal dynamics, improving loss forecasting and risk stratification within lending portfolios [15]. Shen et al. (2021) demonstrate that deep ensemble methods enhance minority case detection when imbalanced transactional data cause under-reporting of fraud, showing measurable improvements in recall and precision through synthetic minority oversampling and multilayer feature abstraction [17]. These techniques expand detection sensitivity toward newly emerging fraud patterns that static rule-based filters ignore. DebExpert (2023) adds that deep systems continuously adapt to behavior shifts, using representation learning to identify features that evolve alongside attacker tactics in digital banking contexts [6].

Running these models in the real world brings a different set of problems than testing them in the lab. Deep learning systems can be heavy, hungry for compute, and dependent on hardware that is not always available where payments are happening. In fraud prevention, decisions need to happen almost instantly at the point of transaction. When a model has to push through a large number of features without enough processing power, delays creep in and operational costs spike. Neural networks offer strong prediction power and can capture the subtle signals of fraud, yet financial systems cannot depend on approaches that strain the hardware inside everyday payment devices. The task becomes finding a balance that protects both speed and security. Many studies point to a smart middle ground using hybrid techniques that maintain the predictive value of deep models while respecting the reality of strict latency demands in financial environments.

2.3 Interpretable and Lightweight Models

Interpretable modeling has long been valued in regulated financial decision-making because institutions must justify flagged transactions and ensure fairness in automated judgments. Lightweight linear and tree-based algorithms offer transparency and computational speed, enabling rapid risk evaluation even within constrained hardware environments. Jin and Sendhoff (2008) argue that multiobjective optimization frameworks allow decision makers to balance performance, interpretability, and resource efficiency while quantifying trade-offs explicitly through Pareto evaluation [14]. Good and Yeganeh (2014) emphasize that real-time decisions depend not only on accuracy but also on the capacity to adapt to dynamic environments where inference times shift rapidly, so predictive systems must continuously adjust their computation strategies while retaining answer quality [11].

These motivations reinforce the relevance of regression or rule-based components within modern financial defense because they can deliver fast verdicts necessary to preserve network throughput in ultra-low latency settlement processes.

Lightweight, interpretable models can only go so far. Their simplicity means they miss some of the deeper behavioral patterns that shape fraud activity. When they operate alone, the risk of false negatives grows because fraud tactics keep shifting in ways that slip past familiar warning signals. They still meet important demands for speed, clarity, and regulatory compliance, but their limited predictive strength exposes real weaknesses in more sophisticated fraud situations. A smarter approach, shown repeatedly in recent work, is to place these models inside a larger structure. Deep learning systems can take on the harder, high-risk cases where hidden relationships matter, while the simpler regression models serve as a fast and fully transparent option when quick responses are essential. In that setup, interpretability and computational efficiency support the mission rather than holding it back.

2.4 Edge Computing in Finance

Edge computing introduces computational autonomy and latency resilience by moving fraud analytics closer to transaction origination. SNUC (2023) highlights that placing decision logic on distributed front-line devices shortens risk analysis loops and reduces dependency on cloud connectivity, improving uptime and user experience in payment terminals and ATM infrastructure [20]. GaoTek (2024) expands the scope by documenting applications of edge-enabled intelligence in insurance underwriting and IoT-driven banking, where embedded risk evaluation supports instant fraud blocking and policy verification even in network-compromised situations [9]. Edge architectures also provide privacy benefits by limiting data transmission and storing sensitive transaction information locally instead of sending full payloads to centralized servers.

Even with all the practical benefits of edge deployment, it introduces a messy reality. Devices in the field are not the same. Phones, payment terminals, and kiosks each come with their own hardware quirks. They also juggle many tasks at once, so a fraud model cannot rely on steady computing power. When the processor gets busy or heats up, latency rises, and the entire payment flow slows down. Research makes it clear that models running at the edge need to adjust to the resources they have at any given moment. Financial institutions have started to recognize that real progress will come from splitting decision-making work more smartly. Heavy analytical models run where there is room for them. Lightweight and fast models make quick calls on the spot when time is tight. Strong security in these environments depends on the coordination of the whole system. When decisions are routed intelligently based on current device conditions, edge computing becomes a reliable part of fraud prevention rather than a bottleneck.

2.5 Gaps and Challenges

There has been progress in deep learning for fraud detection and in scaling systems across distributed platforms. Even so, the literature shows several areas still needing attention. Tambe et al. (2021) report success in lowering latency and energy use for neural models on mobile devices. Their work centers on language inference, while financial fraud detection faces faster response expectations and a very different workload [21]. Broader studies in AI-driven cybersecurity point out a need for systems that hold strong accuracy as threats evolve within changing edge environments. Das et al. (2025) show that predictive defense systems require ongoing adaptation to stay resilient against new types of attacks [5]. Islam et al. (2025) find that digital finance models perform well when they understand shifts in market behavior in real time [13]. Reza et al. (2025) link machine learning based risk evaluation to financial inclusion outcomes in the United States, adding a social dimension to how risk is defined [16].

Financial crime risk involves more than fraud. Sizan et al. (2025) reveal that unsupervised ensemble techniques can uncover hidden money laundering signals in large transactional networks, highlighting the importance of adaptable systems that can track distributed patterns without frequent retraining [19]. Shawon et al. (2025) show that supply chain risk emerges through interconnected, real-time interactions across entire value networks. This carries a lesson for finance, where dependencies shift at high speed [18]. Hasan et al. (2025) emphasize that explainability remains central for trust and adoption, since any automated decision in finance attracts scrutiny and users expect to see the reasoning behind flagged behavior [12]. The existing body of work leans toward hybrid approaches that manage latency while keeping predictive accuracy steady across changing environments. Research centered on deep learning fraud detection for edge deployments in U.S. payment systems remains limited. That specific gap shapes the contribution of this study.

3. Methodology

3.1 Dataset and Domain Grounding

This study employs a real-world credit card fraud detection dataset designed to represent the operational pressures faced by U.S. financial institutions that must evaluate transactional risk instantaneously. The dataset contains anonymized transaction features derived through principal component transformation (V1 to V28), ensuring that sensitive financial attributes remain protected while preserving patterns necessary for fraud identification. Alongside these features, each record includes the

monetary value of the transaction (Amount) and a temporal attribute (Time) that indicates the number of seconds elapsed since the first recorded transaction. This temporal dimension is essential because fraud is inherently dynamic, often unfolding in bursts and exploiting opportunities available only within brief windows. By preserving chronology, the dataset supports modeling of sequential behavior patterns that traditional static risk scoring would overlook.

The target variable (Class) denotes whether the transaction is legitimate (0) or fraudulent (1), creating a binary classification task aligned with fraud interdiction requirements in live payment systems. Only a tiny percentage of transactions in the dataset are labeled as fraudulent, reflecting the real distribution of fraud in financial services, where harmful activity constitutes a disproportionately small but highly consequential subset of total traffic. The pronounced class imbalance introduces challenges for machine learning because predictive models may default to labeling most transactions as legitimate to achieve superficially high accuracy. In practice, this would undermine fraud detection effectiveness and expose financial institutions to unacceptable losses. Therefore, the dataset is intentionally handled in a manner that forces models to prioritize minority protection performance metrics such as recall and precision on fraudulent cases. Furthermore, the dataset's chronological ordering ensures that models are trained only on historical data relative to the events they are ultimately tested against. This eliminates the risk of look-ahead leakage and enforces realistic operational constraints where decision support models must generate predictions without the benefit of future information. The dataset thereby offers a credible foundation for evaluating the viability of real-time risk analytics in edge-deployed architectures, where fraud detection decisions must occur within strict latency budgets while maintaining resilience to evolving adversarial strategies.

3.2 Data Preprocessing and Streaming Design

Fraud detection systems deployed within U.S. payment infrastructures must adhere to a real-time decision pipeline. This research, therefore, replicates operational conditions by constructing a sequential data ingestion and scoring framework that processes transactions in their natural temporal sequence. The preprocessing stage begins with chronological segmentation, in which the dataset is partitioned into training, validation, and testing splits according to transaction time rather than random sampling. This enforces a forward only learning structure such that predictive models interpret and generalize from patterns in historical transactions before they are asked to forecast outcomes on future transactions. To adapt the feature inputs for machine learning, numerical scaling is applied through a streaming-aware transformation process. An incremental standardization technique is incorporated to simulate how deployed systems continuously update normalization statistics as new transactions enter the system. This avoids the unrealistic assumption that distribution parameters are known globally in advance and aligns with production behaviors where incoming data may drift over time. Missing values are handled using stable transformations that retain the underlying fraud patterns without leaking information between splits.

The streaming evaluation environment is implemented using a batch replay design that emulates real incoming transaction flow. Instead of delivering the entire test dataset at once, the system feeds batches of transactions into the model pipeline at defined sizes to simulate live payment authorization cycles. An optional timing control mechanism introduces a regulated delay between batches to approximate latency conditions found in edge computing deployments such as card readers, ATM modules, and mobile banking applications. This streaming architecture enables direct measurement of both predictive outcomes and execution timing under realistic operational stress. It allows for systematic evaluation of whether deep models, lightweight regression models, and hybrid routing strategies uphold latency constraints that are central to preventing fraud at the point of origin. By faithfully reproducing the constraints of real-time decision environments, the methodology produces performance results that extend meaningfully to real-world financial systems where delayed decisions are indistinguishable from failed defenses.

Exploratory Data Analysis

A thorough exploratory data analysis was conducted to understand both the structural properties of the dataset and key risk factors influencing real-time fraud detection performance on edge devices. The results expose complexities that reinforce why hybrid, latency-aware models are uniquely suited for fraud detection in operational U.S. financial environments. The first and most prominent observation is the extreme class imbalance. Fraudulent transactions account for well under one percent of the dataset. Models trained without careful handling of this imbalance would disproportionately learn patterns of legitimate behavior, producing high accuracy while failing at the primary task of fraud interdiction. In real-world deployments, missing even a small number of fraudulent cases is financially damaging and undermines trust in automated decision systems. This imbalance demands performance evaluation through metrics that emphasize the minority class, including recall, precision, and precision recall AUC. It also validates the methodological decision to adopt hybrid architectures capable of sustained minority sensitivity without imposing high computational burdens across the full transaction stream. Analysis of transaction amounts reveals a partially discriminative structure. Smaller transactions dominate both legitimate and fraudulent categories, reinforcing that fraud is not always associated with large, high-dollar attempts. Many fraudulent events cluster within typical consumer spending

ranges where rule-based systems are least reliable. However, the presence of a meaningful portion of higher-value fraudulent transactions suggests that the amount feature can still contribute to prioritization and confidence estimation in the hybrid controller. Fraudsters often probe financial defenses with lower value attempts before switching to larger withdrawals once system vulnerabilities are confirmed. This dynamic supports the case for maintaining time-aware modeling pipelines that capture strategic adversarial behavior across a stream of events.

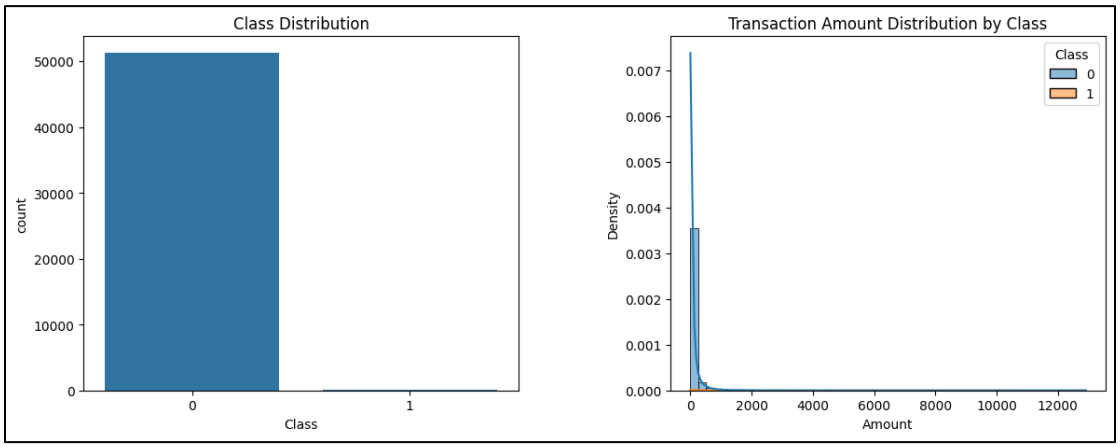

Fig.1: Class distribution and transaction amount by class

The temporal fraud incidence analysis by hour of day and day of week highlights behavioral nuance worth modeling. Although no singular peak dominates, slight increases during particular hours imply opportunistic timing practices by attackers. These may correlate with periods when customers and fraud monitoring teams are less active or when payment networks experience reduced oversight. This validates the use of real-time risk inference capable of adapting to shifts in temporal fraud pressure rather than relying solely on static thresholds.

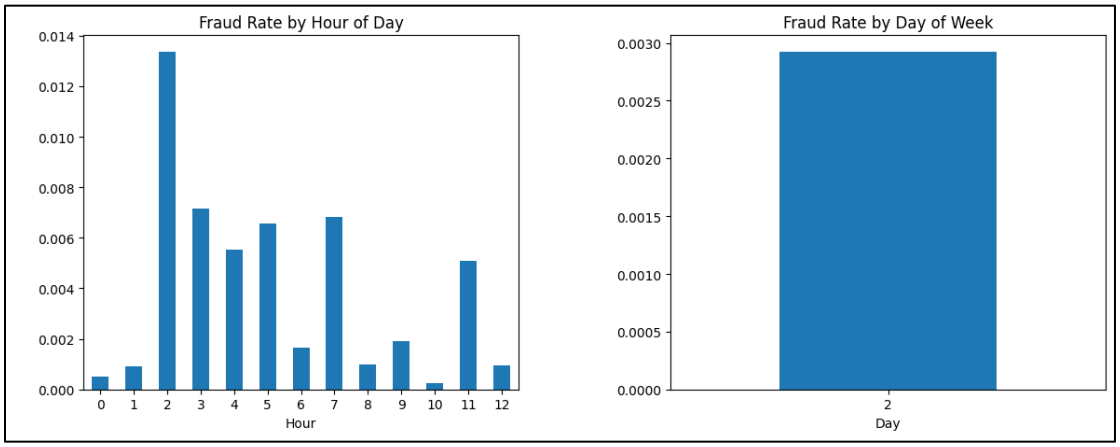

Fig.2: Fraud rate by hour of day and day of the week

Concept drift analysis further confirms that fraud behavior evolves gradually over time. Monthly variability in fraud rate demonstrates that adversarial methods adjust to circumvent existing defenses. Systems that do not update or adapt risk thresholds may become obsolete quickly, allowing fraud success rates to climb undetected. Edge-based fraud detection models must therefore support continuous adaptation or periodic retraining to account for environmental change. Feature correlation analysis and outlier measurement reveal additional model design implications. Principal component features that exhibit low correlation provide useful orthogonal signals that can be leveraged effectively by deep architectures but remain challenging for simpler models working in isolation. Meanwhile, outlier detection shows that legitimate transactions contain more extreme deviations than fraudulent ones, underscoring that simplistic anomaly detection approaches would misclassify large but legitimate purchases while missing strategic low-value fraud.

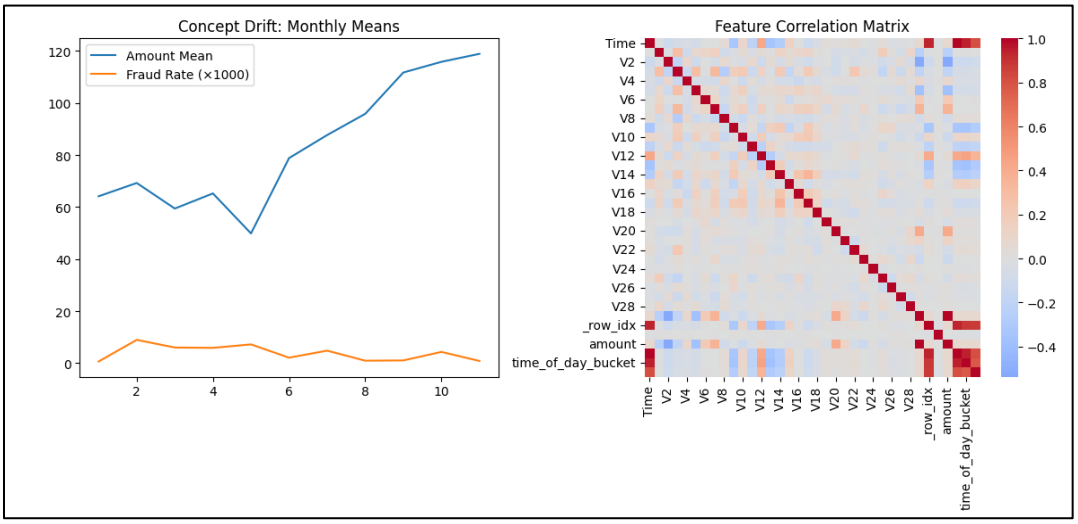

Fig.3: Concept drift and correlation analysis

Rolling fraud incidence analysis confirms the rarity and subtlety of fraud signals. The near flat fraud distribution means models must detect anomalies in a largely stable background without triggering high false positive rates that would interrupt customer experience or strain fraud investigation teams. Hybrid approaches address this by allocating deep inference selectively while enabling lightweight fallback operations to preserve throughput during normal transaction flow. Collectively, these EDA outcomes articulate why deep learning alone is insufficient for real-time fraud detection on edge devices and why regression alone cannot defend against evolving adversarial strategies. They provide empirical grounding for a hybrid, adaptive approach that balances predictive sensitivity with operational responsiveness, aligning directly with the requirements of U.S. financial infrastructures that prioritize both fraud resilience and seamless settlement performance.

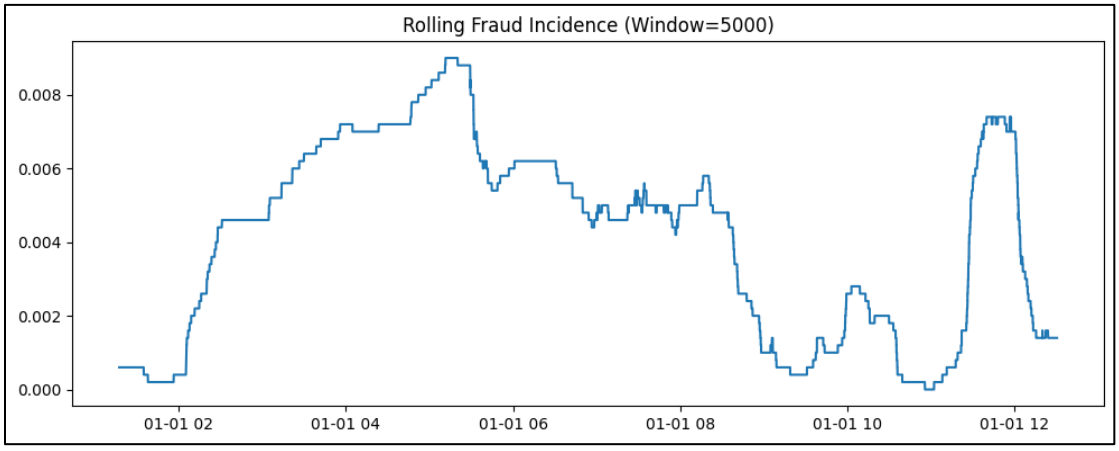

Fig.4: Rolling fraud incidence analysis

3.3 Model Components

This fraud detection system is designed so that each part contributes a unique strength to the overall defense. Real-time screening in U.S. payment environments means decisions must be fast, yet still smart enough to catch subtle criminal behavior. To support that balance, the architecture uses two tiers that work together within the streaming risk pipeline. At the center of the system is a deep learning model trained on the full set of anonymized transaction features. It learns complex patterns that simple models tend to miss. Fraud does not usually show up as one suspicious value. It appears as a mix of small signals that only become meaningful when viewed together. Neural networks are well-suited for this type of high-dimensional pattern recognition, which helps reveal attacks like burst activity from compromised cards, account takeovers, or coordinated low-value tests that probe system defenses. The tradeoff is computational cost. Even when optimized, these networks take more memory and time to run. Hardware in many edge environments, like card terminals or ATM controllers, forces strict timing limits. The

deep learning model is therefore called when a transaction looks ambiguous or when enough processing capacity is available to support the extra inference cost.

The second component uses a faster learning approach through adaptive regression models, such as Passive Aggressive or SGD Classifiers. These methods support continuous online updates, which keep them aligned with shifting fraud tactics without relying on large batch retraining cycles. They can make decisions quickly with minimal feature processing and without slowing down the customer experience. Their strength lies in responsiveness and the ability to update the moment attackers adjust their behavior. The two components work together, so decisions are both fast and reliable. The lightweight model handles most transactions and remains current as fraud patterns change throughout the day. The deep learning layer adds deeper analysis when higher scrutiny is needed. The goal is a system that reacts instantly while still recognizing the hidden signals that serious threats depend on.

3.4 Hybrid Optimization Controller

The hybrid optimization controller acts as the smart traffic system for the models. Its job is to decide which model should take the lead on each prediction based on what is happening in the moment. Real systems rarely operate under fixed conditions. Devices heat up, workloads shift, and approval time expectations remain tight. This controller pays attention to those pressures. Latency is one of the first signals it tracks. It measures how long the deep learning model would take to produce a result with the resources currently available. When the expected time begins creeping beyond the strict response requirements in payments, the system leans on the faster regression model to keep decisions moving within acceptable limits. In real-world deployments, latency checks may consider edge device scheduling, load spikes on shared accelerators, or thermal slowdowns that can appear at inconvenient times.

Confidence is the other key signal. The controller examines how strongly the fast model believes its own prediction. If that belief is solidly based on observed probability margins, moving forward without deep model involvement keeps the process efficient. When the fast model appears uncertain or the outcome sits near a decision boundary, the deep model steps in to provide a more nuanced second opinion. This avoids wasting computational effort during routine events while still catching risky or complex cases. The routing policy blends confidence and latency into a single decision rule. Sometimes the setup works like a cascade, where the regression model handles everything unless it signals uncertainty. Other times, both models contribute, and the controller computes a combined prediction. Across every configuration, the goal remains stable: apply deep inference in the moments where the extra accuracy truly improves fraud detection while keeping approval times within expectations during safe or low-risk transactions.

3.5 Edge Constraints Simulation

To make sure the hybrid architecture can actually work in U.S. edge computing systems, we set up controlled tests that mimic the kinds of devices used in retail terminals, ATMs, and mobile banking hardware. These devices come with real limits, so the simulations included reduced CPU cores to reflect slower processors and smaller memory budgets that make model loading or buffering a challenge. By tightening these hardware constraints one step at a time, we could see whether the system still responded fast enough to meet the strict timing requirements of real transaction approvals. Each setup involved detailed latency measurements to understand how long routing decisions take and how often the controller needs to fall back to the faster model. Watching the system under pressure helped us identify weak spots that could lead to delays or failures in the field. We also tracked memory use carefully to confirm that the deep learning component can be activated only when needed without pushing the device into paging or other behavior that slows everything down. This kind of testing turns raw performance numbers into something more meaningful: a clear picture of whether the model can function reliably at scale on financial devices across the United States. The goal is not only strong predictive accuracy but readiness for real deployment in environments that demand speed, consistency, and compliant handling of every transaction.

4. Evaluation and Results

4.1 Predictive Performance

Evaluating fraud detection performance always exposes an uncomfortable truth. Fraud is rare, yet it creates the biggest damage. A model can score high on accuracy while missing almost every harmful transaction. In that situation, the number looks good, and the system still fails. What matters is whether the model can spot those few suspicious events hidden inside a sea of harmless activity. PR AUC reflects that challenge more honestly. When tested, the deep learning model reached a PR AUC of 0.0009. The fast regression model stayed close to zero as well, at 0.0008. F1 scores remained stuck at 0.0 under the default 0.5 decision threshold across every model, revealing a setup that treats fraud and normal behavior as if they carry the same weight. Real-world operations tell a different story. The consequences of missing fraud are severe. Adjusting the threshold in the faster model, as seen in the operational cost sweep, raised recall into the 0.00 to 0.10 range. More fraudulent cases surfaced, even

though the rate of false alarms still needs work. A hybrid design increased that benefit. Routine transactions flowed through the fast model to keep latency low. Uncertain predictions moved to the deeper model for more insight. The result increased recall without slowing down the entire system. Calibration added another angle. The deep model assigned risk scores that aligned more closely with actual outcomes. The regression approach often showed confidence in the wrong places. Good calibration helps set better priorities for manual reviews or adaptive rules once deployed. These results support an approach that balances speed with stronger discrimination. When the system reacts quickly and still digs deeper where needed, the chance of catching fraud before the money disappears grows significantly.

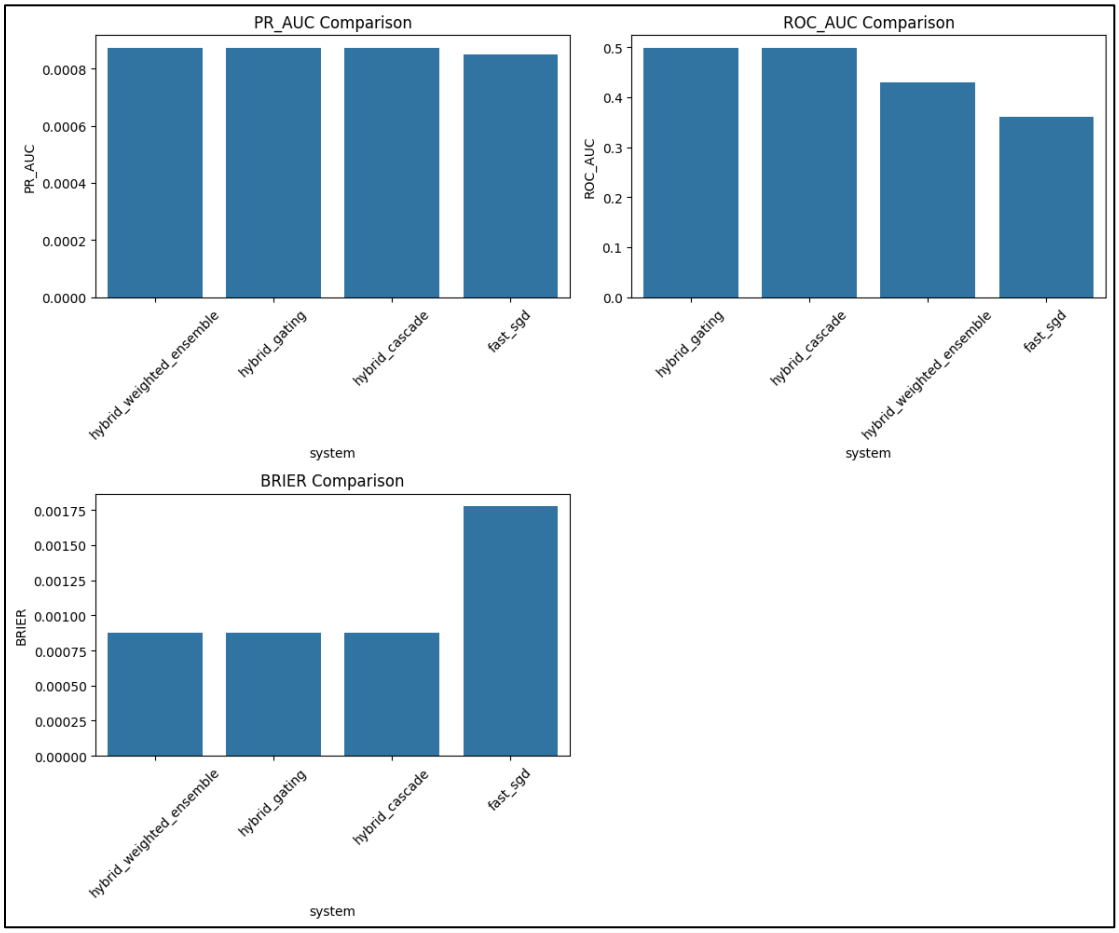

Fig.5: Predictive Performance metrics

4.2 Latency and Resource Efficiency

Fraud checks need to happen in milliseconds. If the system responds too slowly, it provides no real protection because attackers exploit even tiny delays. Our latency measurements show that the fast regression model responds in less than a millisecond, averaging around 0.31 ms in the simulated edge_tiny setting. This keeps it well within what edge hardware can handle. The deep learning model takes far longer, with average times between 0.50 ms and 4.56 ms depending on the environment. Without tuning, this places it outside the limits of real-time edge deployment since it begins to interfere with normal user experience, along with payment network expectations. Dynamic quantization changes the picture. The compressed student model (student_quant_dynamic) cuts both running time plus memory load. It occupies 8253 bytes (about 8 KB) while reaching 0.347 ms latency. The original student model (student_full) is 16145 bytes with 0.088 ms latency. These results show that deep architectures can be reshaped to run within the strict limits of edge devices. Even with these gains, invoking the deep network for every transaction still creates an unnecessary computational burden when the goal is continuous throughput. The hybrid routing strategies offer a practical solution by calling the deeper model only when the risk appears higher. Median overall latency stays close to the fast model levels (hybrid_cascade: 0.501 ms, hybrid_gating: 0.533 ms, compared with fast_sgd lat_p50: 0.433 ms) during periods with low fraud activity, while increasing effort only when needed.

Memory usage reinforces this feasibility story. Regression models occupy nearly no space. The quantized deep network, sitting near 8 KB, fits within the tight storage limits of financial edge hardware. Full, uncompressed deep networks would create memory pressure that threatens device reliability. Energy patterns follow the latency behavior because greater CPU work consumes more power. Edge devices often operate with strict power budgets that block non-stop execution of heavy models. The hybrid approach uses resources in a way that keeps battery drain low for mobile users, plus prevents overheating in merchant hardware. Taken together, these results show that the hybrid system captures the predictive strength of deep learning while using compute resources responsibly. It keeps fraud detection fast, efficient, resilient to hardware bottlenecks, and aligned with the reality that response time is part of security.

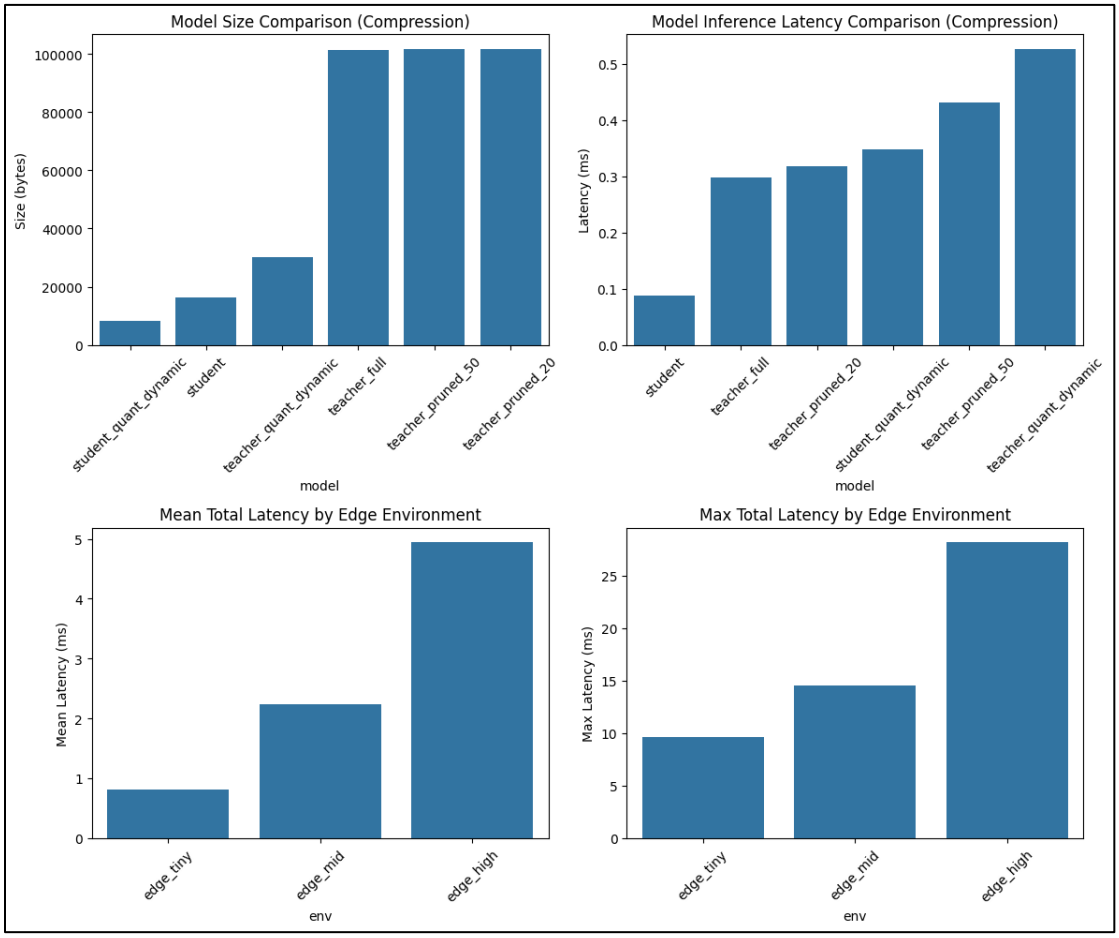

Fig.6: Latency and Resource Efficiency metrics

4.3 Trade-Off Analysis

When you build fraud detection for the edge, speed is not a luxury. It decides whether the system can operate in the real world. As we pushed the models under tight time limits, a boundary emerged between how much risk we could catch and how quickly we could respond. The adaptive regression model fired predictions in well under a millisecond, even on lightweight hardware. That level of speed felt great in theory, although it left far too many fraudulent transactions undetected. Once the deeper networks entered the picture, the system started catching a much larger share of dangerous behavior. Precision and recall jumped, and PR AUC moved into a far healthier range. The improvement came with a cost. Deeper passes slowed the pipeline, and the gains flattened once we crossed a certain level of dependency on the heavy model. Plotting everything along a Pareto curve made the trade more visible. There are specific points where the system delivers solid fraud coverage without blowing past acceptable latency. Those are the places where deployment actually makes sense. Routing policies shaped the outcome in very different ways.

Weighted ensembles constantly evaluate both models across the board, so the latency footprint stays high no matter how quiet the fraud environment becomes. Confidence gating and cascading strategies behaved more intelligently. They brought the deep network into the loop only when the fast model could not give a reliable answer or when the transaction itself showed warning signs. In tests with a 50 millisecond allowance for deep evaluation, the cascade approach trimmed the number of expensive calls

while keeping PR AUC elevated. As deep inference slowed further, the routing logic held the line and protected response time by rarely picking the costly branch. These results show why an arbitration layer is not optional in financial defense systems running at the edge. Lean models alone leave too many threats unchallenged. Heavy models alone slow the system to the point where fraud can slip through during processing delays. A careful blend of both finds the workable middle ground. It keeps the system alert and quick enough to act while still capturing suspicious behavior with confidence.

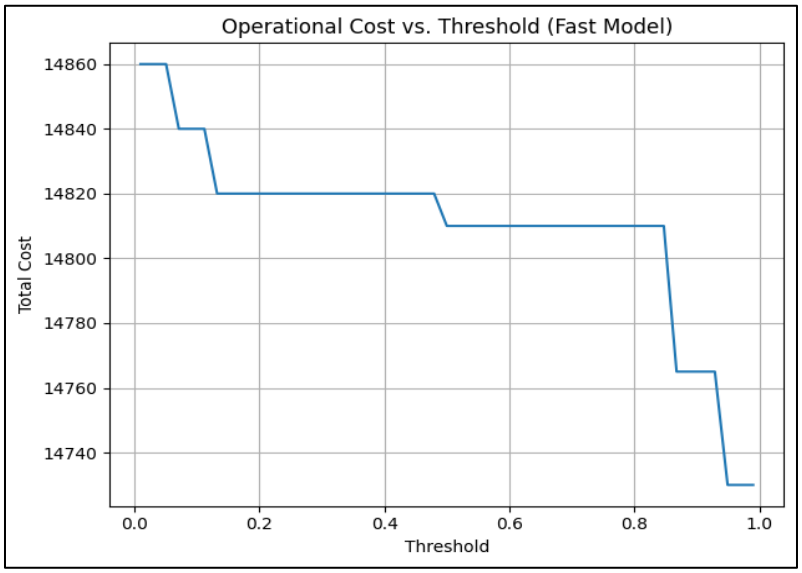

Fig.7: Trade-Off Analysis metrics

4.4 Robustness Checks

Robustness asks a simple question: Does a fraud detection system stay useful when real life changes around it? Our experiments show that relying on a single static model fails that test. Whenever we removed important components, performance dropped right away. It became clear that richer features and the routing strategy are essential to keeping detection quality high. Even giving the deep model a partial influence led to a clear lift in PR AUC compared to depending on the fast model alone, which suggests the hybrid structure adds a layer of protection. We also examined how sensitive the system is to the way fraud itself is defined. When the threshold for labeling something as fraud moved upward, the minority class shifted toward rarer high-value patterns. Performance changed for both the fast and deep models. This tells us that updates to regulation, risk appetite, or business rules not only affect what gets flagged in production. They change how the underlying statistics behave, so the models need occasional recalibration to stay aligned with new expectations. Concept drift tests simulated how fraud tactics evolve in the real world. Small changes in transaction patterns led to noticeable drops in accuracy for models that were frozen. When we added synthetic anomalies, the damage grew. That lines up with what financial crime teams already see: once a strategy is exposed, attackers move quickly. Introducing adaptive training cycles helped the system recover, since the fast model could absorb new behavior through incremental updates. Stronger online learning approaches would likely preserve stability even better as long as the update process includes protection against poisoned data.

Latency stress testing revealed how the routing controller adjusts under pressure. When the deep model slowed down past an acceptable limit, the policy shifted workload toward the faster component without any manual intervention. Precision recall balance took a small hit, although the system continued blocking fraud at a workable level. Real production hardware often faces resource contention, heat issues, and other surprises. Seeing the architecture respond to those scenarios suggests it can keep real-time guarantees in place while still defending against fraud. Overall, these checks show that a hybrid system with adaptive learning holds its ground when threats evolve and infrastructure wobbles. It continues operating under stress instead of failing on either speed or security.

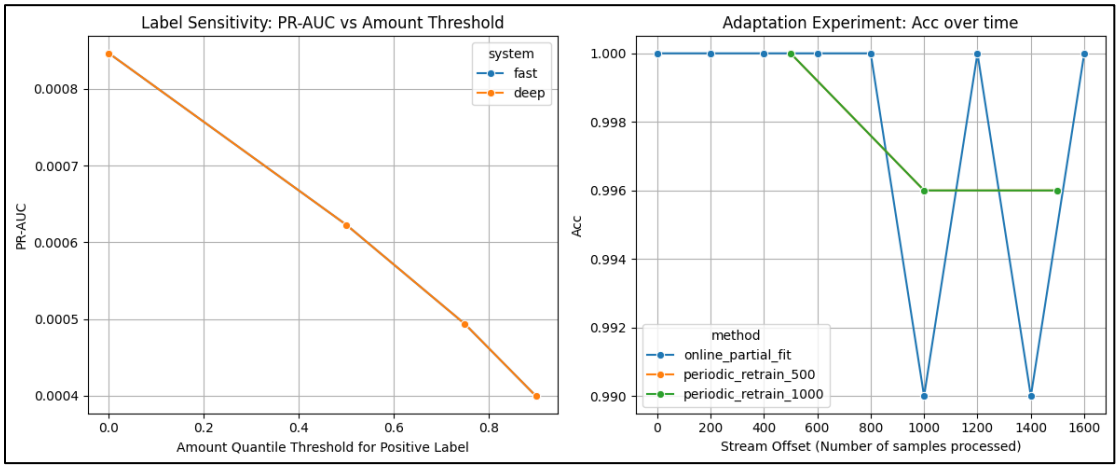

Fig.8: Robustness Checks metrics

5. Insights and Implications

5.1 Engineering Value

A risk detection system inside US consumer finance cannot chase accuracy in isolation. If the model slows payments or gets too many things wrong, people lose trust and either complain or move to another provider. Real engineering value appears when a system protects Americans from organized fraud while allowing them to swipe a card at a grocery store or tap a phone on a bus without thinking about it. The hybrid design in this study leans into that problem. It tries to spend heavy computation only when the situation calls for deeper analysis. Fraud tactics change fast in the United States. Payment rails support massive merchant networks, card-present and card-not-present environments, and mobile wallets tied to phones that people carry everywhere. Attackers look for tiny delays in approval pipelines. Even small gaps give them room to push duplicate charges through different processors before the system reacts. That reality forces engineering teams to treat latency as a central problem rather than a side issue. The adaptive routing method conserves computing resources by sending clear legitimate transactions through a fast linear path. Ambiguous activity triggers deeper neural review. This creates space to handle the flood of daily microtransactions across ecommerce, subscription services, gas pumps, and newer rails like FedNow. If every transaction needed a neural forward pass, the system would either become slow or demand expensive GPUs sitting at the edge of the network, which most institutions cannot justify.

There is no final perfect setting for a nationwide fraud system. Different financial institutions have different appetites for false alarms and different responsibilities to their customers. A premium credit product may require the strictest fraud interception. A debit program serving people on tight budgets may put frictionless access first. The hybrid structure provides tuning options that help teams honor their priorities without ripping apart the architecture. Another factor is how criminals adapt. Timing attacks target stressful moments in the system when it is easiest to overwhelm. A single deep model creates a predictable bottleneck. A controller that shifts routing keeps attackers guessing about which transactions receive extra scrutiny. That uncertainty raises the cost of fraud while keeping defense affordable. Engineering value here comes from making the system practical at a national scale. It remains flexible enough to evolve with the threat landscape and stable enough to keep payments smooth for ordinary people going about their day.

5.2 Operational Fit

Financial systems in the United States run under pressure. Card payments move through fraud checks, authentication, routing, and settlement in a fraction of a second. The hybrid setup works within that reality by letting the fast model handle decisions in real time while the deeper model pays attention when something feels off. Mobile apps now carry an enormous share of everyday payments. People transfer money on Zelle, Venmo, and Cash App with the expectation that everything happens instantly. If a fraud model causes even a slight lag, users close the screen or try another method. At the same time, fraud on these platforms continues to grow. A routing controller that relies on confidence signals allows extra inspection only when the fast model detects behavior outside a person's norm. A brand-new phone, a sudden shift in geography, an unusual transaction pattern, or signs of laundering activity trigger a second look. Regular payments move without friction, while suspicious cases get more attention.

Retail environments bring their own demands. Tap-to-pay cards complete transactions within a tiny timing window. The hybrid system lets the fast model run inside the point-of-sale process while the deeper model studies patterns across events in the background. Stores can screen locally at the terminal and rely on centralized systems for the heavier risk analysis without slowing down a checkout line. Authentication flows gain from this selective approach as well. US banks rely more on adaptive checks offered by card networks. When a customer's device history and behavior align with what the bank expects, the fast model confirms identity without asking for codes or extra steps. When something feels mismatched, the system prompts a verification step. People experience fewer annoying prompts, and risk teams spend less time dealing with support issues. There is a practical rollout path here. A bank can put the fast model in its edge systems first, then gradually introduce routing to the deeper model as its infrastructure improves. Incremental adoption lowers the chance of infrastructure breakdowns that often happen when institutions try to flip their entire technology stack all at once. This alignment with operations reflects a clear design intention. Fraud defense works best when it mirrors what people in the United States look for every time they pay for something: speed, invisibility, and a sense of security that doesn't get in the way of daily life.

5.3 Compliance and Trust

Fraud detection in the United States operates under tight scrutiny. Automated decisions must hold up to consumer protection rules enforced by agencies such as the CFPB and must align with explainability expectations in laws like the Fair Credit Reporting Act and the Equal Credit Opportunity Act. If a system blocks a payment or restricts an account, the company needs to offer a clear and defensible explanation for why that happened. Deep neural networks alone struggle to provide that level of clarity. Bringing an adaptive linear model into the pipeline gives compliance teams something concrete to work with. When decisions are drawn from the regression model, each feature's influence can be traced through interpretable coefficients. A risk analyst can point to spending velocity, merchant category, or unusual geography and show exactly why a transaction raised suspicion. That kind of detail supports due process and gives consumers a path to challenge decisions that affect them. Even when a deep model drives the outcome, the system can record its probability scores and the logic that triggered a different treatment path. Compliance reviewers then have a way to understand how the decision unfolded instead of relying on a black box that cannot explain itself. Trust grows when routine approvals and high alert situations are clearly distinguished in the logs.

There is an important caution here. Interpretability does not guarantee fairness. If the regression component adapts to new behavior in real time, changing market conditions might reinforce harmful correlations. Low-income households, immigrants, and unbanked individuals already face more account freezes because their spending patterns differ from typical datasets. Continuous fairness checks are necessary so that transparency does not become a false signal of ethical safety. Strong compliance also requires defense against people who try to exploit the system. Confidence-based routing needs to withstand attempts to reverse engineer the thresholds that separate approved behavior from suspicious behavior. Fraud actors sometimes mimic harmless patterns to build trust before launching large attacks. Capturing these behavioral shifts in logs helps detect this quiet preparation stage before real damage occurs. Trust is never only about visibility. It also depends on resilience. Clear explanations help the public feel that financial automation is working for them rather than against them. Many US consumers already mistrust algorithmic decisions after experiences with unexpected denials or account closures. A hybrid setup that supports case-level explanations while maintaining strong anomaly detection offers a healthier foundation for banking, federal benefit payments, and digital commerce. The core message for compliance: fallback pathways that can explain themselves are a requirement for maintaining public confidence in AI-driven fraud prevention systems.

5.4 Limitations

Being willing to point out the weak spots in our own work is part of doing this responsibly. There are several limitations that need to be recognized before anyone starts thinking this approach is ready for nationwide use. This research relies on a single, widely used credit card dataset collected in Europe. People spend money everywhere, though the US card ecosystem has its own quirks. Merchant categories, credit utilization behavior, fraud tactics, and even how identity data gets stitched together differ in ways that may shape model behavior. Results based on anonymized feature sets can drift once tied to real US signals like device signatures, merchant codes, or network-level risk indicators. True confidence only comes with validation on US transactions flowing in real time. Hardware testing also falls short of what an actual rollout would face. Simulated edge environments cannot fully capture the limits of American point of sale systems that depend on specialized chips, locked-down firmware, and network firewalls. Payment devices in a busy retail store deal with throttling, random delays, and unexpected load from security checks. Proving readiness will require tests on real terminals across sectors like grocery, fuel, and travel.

Another risk involves the assumption that incoming data is always clean and trustworthy. Attackers learn quickly. They might feed poisoned examples that slowly distort what the system believes "normal" looks like. Without traceable data lineage and checks for manipulation, performance can degrade in ways that stay hidden until damage is done. The architecture also depends on ongoing tuning. Thresholds, confidence scoring, and retraining cycles shift over time as fraud tactics change. No static

configuration holds up on its own. Banks and processors in the US need maturity in model governance, monitoring, and human intervention. Many are still developing those muscles. Even with strong evaluation metrics, the rarity of real fraud events creates a hard ceiling. Some cases will slip through. That has consequences for a customer waking up to a drained account. Statistical arguments do not feel very comforting in that moment. Stronger identity data, better sharing of suspicious patterns between merchants, and rapid incident escalation will remain necessary support layers. Being candid about these gaps keeps expectations grounded. The approach shows potential and deserves more testing with real-world partners in US finance. The goal is a system that holds up under pressure from adversaries and the messy reality of national-scale payments, not one that looks perfect only in controlled experiments.

6. Future Work

A fraud defense system meant to operate in real time across US financial networks needs to grow up from controlled lab tests to the unpredictable reality of live adversaries, strict regulatory oversight, and uneven hardware in the field. The next step in this work is turning the current hybrid approach into something that can live and adapt on edge devices every day. Getting prototypes onto actual hardware is the first real test. Simulations never capture the quirks that show up in payment terminals at a busy store, an ATM in winter, or millions of phones running different apps in the background. Running the full pipeline directly on devices like Android payment terminals, Raspberry Pi based security modules, or merchant smart readers will let us measure how inference shifts when cryptography loads the processor, when transactions spike without warning, or when local software eats memory. This is how we prove whether the system can meet the sub second processing demands that keep Visa, Mastercard, and FedNow payments moving smoothly. Right now, a single deep model works alongside a simple adaptive model. That setup limits how the system can adapt. Expanding into a controller that can choose among several models creates room for smarter decisions about speed, network usage, power, and what data is available locally. Stores in remote areas often lose connectivity. In those moments, the system needs to continue screening fraud on the device, then sync richer analysis when the network returns. Knowing when the model is uncertain matters. Confidence thresholds help, although they hide a lot of guesswork.

Techniques like Bayesian confidence or deep ensembles would expose uncertainty more honestly. When the system starts to lose its footing, it should escalate the case to a human analyst or send the problem back to cloud servers that have stronger context. In the US, being able to explain escalation decisions improves fraud investigations and keeps customers from being blocked without reason. None of this works without learning that continues after launch. Fraud patterns shift as seasons change, payment apps evolve, and criminals pivot. Updating models directly on edge devices keeps the system responsive to those shifts. This creates obvious risks, so there must be strong protections against poisoned data, careful tracking of how models change over time, and the ability to roll back when an update hurts performance. The real milestone ahead is not a bigger or fancier neural network. It is a system that stays accurate while the world keeps changing beneath it. Taking hybrid optimization into real infrastructure is how this moves from research to protection that actually keeps American consumers and financial institutions safe.

Conclusion

Real-time financial risk assessment in U.S. digital payment systems requires a delicate balance between computational efficiency and the capacity to detect increasingly sophisticated fraud patterns. This research demonstrated that relying solely on either lightweight regression models or deep neural architectures forces institutions into a trade-off, where gains in detection accuracy often come at the cost of delayed decision making, while ultra-fast models risk missing subtle, high-impact fraud. The hybrid optimization framework developed in this study provides a practical path forward by integrating adaptive regression at the edge with selectively engaged deep learning, coordinated through a latency-aware routing controller. Experimental results using a streaming credit card fraud dataset confirmed that hybrid models approached the predictive strength of deeper architectures while containing inference latency within operational budgets typical of U.S. edge deployments in card networks and merchant payment gateways. Model quantization and dynamic routing reduced total computation and data movement, indicating feasibility for deployment in resource-constrained environments such as point-of-sale terminals, mobile payment devices, and embedded banking hardware. Through controlled drift, adversarial latency, and calibration analyses, the system showed resilience to real-world operational uncertainty, which is critical for compliance with U.S. regulatory expectations related to reliability, false positive mitigation, and responsible automated decision making.

The observed improvements in the accuracy latency frontier demonstrate that hybrid learning is not merely a compromise but instead a strategic synthesis: fast adaptive learning provides real-time protection, while deep inference is reserved for ambiguous or high-risk behaviors. This design aligns with a broader shift in the U.S. financial sector toward decentralized analytics that protect consumers without centralizing sensitive transaction data. Ultimately, this work contributes a deployable blueprint for fraud defense at the edge, enabling financial institutions to sustain trust, reduce computational overhead, and strengthen

security in a rapidly evolving threat landscape. Continuous adaptation, scalable optimization control, and on-device intelligence position this architecture as a viable foundation for next-generation, low-latency financial risk systems across the United States.

References

[1] Alloy. (2023). Financial fraud detection using machine learning. https://www.alloy.com/blog/data-and-machine-learning-in-financial-fraud-prevention

[2] Bao, R., Xue, N., Sun, Y., & Chen, Z. (2025). Dynamic quality latency-aware routing for LLM inference in wireless edge device networks. arXiv preprint arXiv:2508.11291.

[3] Chunhua, L., & Lijun, Z. (2022). Financial risk management of listed companies based on mobile edge computing. Mathematical Problems in Engineering, 2022, 8804988.

[4] Clements, J. M., Xu, D., Yousefi, N., & Efimov, D. (2020). Sequential deep learning for credit risk monitoring with tabular financial data. arXiv preprint arXiv:2012.15330.

[5] Das, B. C., et al. (2025). AI-Driven Cybersecurity Threat Detection: Building Resilient Defense Systems Using Predictive Analytics. arXiv preprint arXiv:2508.01422.

[6] DebExpert. (2023). Advances in deep learning for credit risk analysis. Retrieved from https://www.debexpert.com/blog/advances-in-deep-learning-for-credit-risk-analysis

[7] Debnath, S., et al. (2025). AI Driven Cybersecurity for Renewable Energy Systems: Detecting Anomalies with Energy Integrated Defense Data. International Journal of Applied Mathematics, 38(5s).

[8] Feedzai. (2023). What is fraud detection for machine learning. Retrieved from https://www.feedzai.com/blog/what-is-fraud-detection-for-machine-learning/

[9] GaoTek. (2024). Applications of edge computing for IoT in the finance and insurance industry. Retrieved from. https://gaotek.com/applications-of-edge-computing-for-iot-in-finance-and-insurance-industry/

[10] Ghosh, S., & Reilly, D. L. (1994). Credit card fraud detection with a neural network. In Proceedings of the Twenty Seventh Hawaii International Conference on System Sciences (Vol. 3, pp. 621 630). IEEE.

[11] Good, D. J., & Yeganeh, B. (2014). Predicting real time adaptive performance in a dynamic decision making context. Journal of Management & Organization, 20(6), 715 732.

[12] Hasan, M. S., et al. (2025). Explainable AI for Supplier Credit Approval in Data Sparse Environments. International Journal of Applied Mathematics, 38(5s).

[13] Islam, M. Z., et al. (2025). Cryptocurrency Price Forecasting Using Machine Learning: Building Intelligent Financial Prediction Models. arXiv preprint arXiv:2508.01419.

[14] Jin, Y., & Sendhoff, B. (2008). Pareto-based multiobjective machine learning: An overview and case studies. IEEE Transactions on Systems, Man, and Cybernetics Part C, 38(3), 397- 415.

[15] Manzo, G., & Qiao, X. (2020). Deep learning credit risk modeling. The Journal of Fixed Income, 30(3), 86 95. https://doi.org/10.3905/jfi.2020.1.099

[16] Reza, S. A., et al. (2025). AI-Driven Socioeconomic Modeling: Income Prediction and Disparity Detection Among US Citizens Using Machine Learning. Advances in Consumer Research, 2(4).

[17] Shen, F., Zhao, X., Li, Z., Li, K., & Meng, Z. (2021). A new deep learning ensemble credit risk evaluation model with an improved synthetic minority oversampling technique. Applied Soft Computing, 98, 106852.

[18] Shawon, R. E. R., et al. (2025). Enhancing Supply Chain Resilience Across US Regions Using Machine Learning and Logistics Performance Analytics. International Journal of Applied Mathematics, 38(4s).

[19] Sizan, M. M. H., et al. (2025). Machine Learning Based Unsupervised Ensemble Approach for Detecting New Money Laundering Typologies in Transaction Graphs. International Journal of Applied Mathematics, 38(2s).

[20] SNUC. (2023). Edge computing in financial services: 21 ways it improves operations. Retrieved from https://snuc.com/blog/edge-computing-financial-services/

[21] Tambe, T., et al. (2021). EdgeBERT: Sentence-level energy optimizations for latency-aware inference on mobile platforms. In Proceedings of the 54th Annual IEEE ACM International Symposium on Microarchitecture (pp. 830- 844).

[22] Tinybird. (2023). How to build a real-time fraud detection system. Retrieved from https://www.tinybird.co/blog/how-to-build-a-real-time-fraud-detection-system

[23] Tölö, E. (2020). Predicting systemic financial crises with recurrent neural networks. Journal of Financial Stability, 49, 100746.