zensar

Machine Learning for Real-Time US Mobile Insurance Fraud Detection

(Mobile Claim Risk Scoring Solution)

White paper

An RPG Company

Executive Summary

Mobile device insurance fraud is increasing in both volume and sophistication, driven by identity manipulation, repeated device misuse, and organized fraud networks exploiting digital claim channels [2], [4], [12]. Industry reports consistently show that traditional rule-based approaches and manual investigations struggle to scale, adapt to evolving fraud tactics, and deliver consistent outcomes in real time, particularly in mobile-first digital ecosystems [1], [11].

This paper presents a production-grade, real-time machine learning system designed for a large US-based client to detect mobile device insurance fraud with high accuracy, explainability, and operational stability. Prior research has demonstrated that machine learning (ML)-based fraud detection significantly outperforms static rules in insurance contexts, especially when leveraging heterogeneous data sources and ensemble-based decisioning [7], [5].

Our solution integrates five heterogeneous enterprise data sources spanning claim history, payment risk signals, identity verification outcomes, alert and watchlist matches, and external fraud intelligence, such as the National Intelligence Crime Bureau (NICB), to engineer 104 features capturing historical behavior, temporal dynamics, identity consistency, and fuzzy-match risk indicators. Rich behavioral representations and relationship-aware features have been shown to be critical for detecting organized and repeated fraud patterns in insurance claims data [13], [14].

Multiple gradient-boosting models were evaluated using time-based validation and automated hyperparameter optimization, with XGBoost selected for its balance of predictive performance, interpretability, and low-latency inference. Gradient-boosted decision trees, and XGBoost in particular, are widely adopted in insurance fraud detection due to their robustness on high-dimensional, imbalanced tabular data and suitability for real-time decisioning [6], [12]. SHAP-based explanations are generated for every prediction, enabling investigator trust, auditability, and regulatory compliance in line with best practices for explainable AI adoption in financial and insurance systems [5], [14].

A key contribution of this work is addressing probability and threshold instability commonly observed in production fraud models. Prior studies show that models with comparable ranking performance (e.g., AUC or F1 score) may produce poorly calibrated probability outputs, resulting in operationally disruptive swings in alert and review volumes when static thresholds are applied [9], [11]. To mitigate this issue, the system applies Beta Calibration as a post-model standardization layer, producing well-calibrated probability estimates that support stable decision thresholds across model versions and retraining cycles. Beta Calibration is a mathematically grounded, empirically validated approach that has been shown to outperform traditional logistic and isotonic calibration methods under distribution shift [8].

Deployed using a cloud-native, serverless architecture based on Azure Functions, the system delivers sub-second real-time scoring at an enterprise scale while maintaining elasticity, resilience, and cost efficiency. Cloud-native, event-driven architectures are increasingly recognized as best practice for real-time fraud detection platforms requiring low-latency inference, horizontal scalability, and operational robustness in regulated financial environments [3], [10].

Business context and baselining global risk module system

Before developing the solution, our US-based insurance customer relied heavily on static rules, heuristic checks, and manual reviews conducted by fraud investigators to detect mobile insurance fraud. These activities were supported by the existing global risk module (GRM). While this rule-based approach was effective in identifying obvious anomalies, it lacked scalability and struggled to adapt to evolving fraud patterns.

Fraud risk signals and rules were authored across multiple fragmented systems, including core claims platforms, payment risk providers, identity verification services, internal alerts and watchlists, and external fraud intelligence repositories such as the NICB. This fragmented architecture made it difficult to construct a unified, real-time view of fraud risk. As a result, some high-risk claims went undetected, while low-risk claims were unnecessarily delayed by manual reviews, negatively impacting the customer experience. The magnitude of this problem was substantial.

As part of the baseline assessment, we benchmarked the GRM system and observed that, on average, it processed approximately 12,000 claims per day in 2024. Of these, only about 6.2% were flagged as potentially fraudulent, while the remaining ~94% were categorized as low risk. However, the business believed that a portion of this unflagged population likely contained suspicious claims that were going undetected by GRM. Identifying additional fraudulent claims within this segment was one of the key business objectives (Motive 1).

Within the 6% of claims flagged as high risk, roughly 35% were routed to the triage team for detailed manual review by fraud experts. The remaining 65% was largely denied for entitlement-related reasons, such as lack of coverage at the time of loss. Of the claims sent for manual review, approximately 74% were ultimately determined to be false positives, placing a significant and unnecessary burden on the triage team. This volume distribution is illustrated in Figure 1.

Consequently, the business sought to reduce incorrect classifications, minimize manual effort associated with false positives, and limit unnecessary scrutiny of legitimate claimants (Motive 2). This analysis revealed that the GRM system achieved a precision of approximately 12.4%, which was set as a benchmark for the new solution. Figure 2 illustrates the methodology used to derive this precision metric.

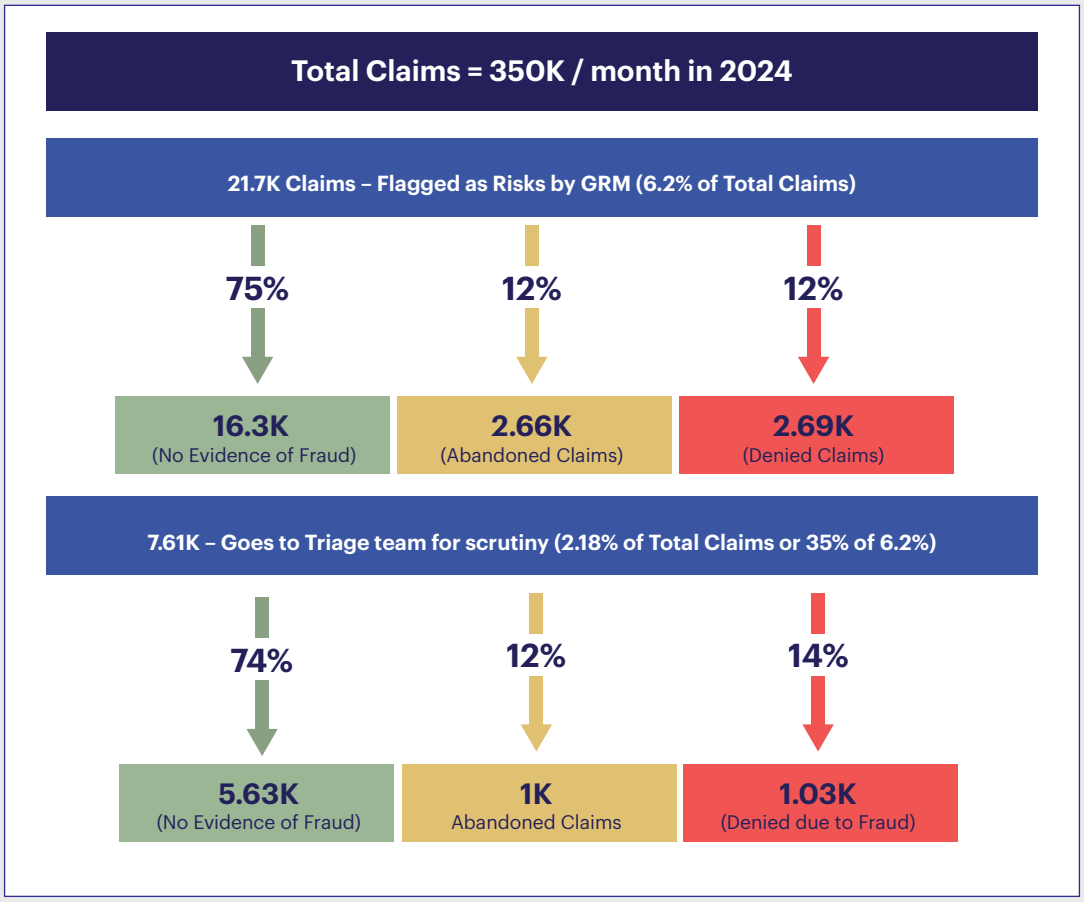


Figure 1: Claim's Volume Analysis

Benchmarking of GRM

CONFUSION MATRIX		Predicted	
		Suspicious	Non-Suspicious
Actual	Suspicious	2,697 (TP) (Denied)	3,28,300 (FN + TN)
	Non-Suspicious	19,002 (FP)	
	Total	21,700	
	Precision	12.4%	

Figure 2: GRM benchmark.

Note: In the above depiction, TP stands for true positives; TN stands for true negatives. FP stands for false positives; FN stands for false negatives.

Beyond these two primary motives, the business team wanted to ensure that our ML solution meets regulatory and audit requirements for explainability and transparency. Even small changes in fraud detection thresholds can materially affect the number of claims routed to manual review, thereby directly impacting staffing requirements, service-level agreements (SLAs), and operational costs. Furthermore, frequent model updates (while necessary to keep pace with evolving fraud) can unintentionally disrupt downstream workflows if model outputs are unstable or hard to interpret. These challenges created a clear business need for a real-time, data-driven fraud detection system that could:

1. Integrate disparate fraud signals into a unified risk score
2. Accurately rank suspicious claims while minimizing false positives
3. Provide transparent explanations for every decision
4. Deliver stable, interpretable probability scores suitable for consistent operational thresholds
5. Operate at enterprise scale with sub-second latency

The solution described in this paper was designed to meet these requirements, enabling faster, more consistent claim adjudication, reducing unnecessary manual investigations, and establishing a trustworthy, scalable foundation for automated fraud decisioning in mobile device insurance.

Solution architecture

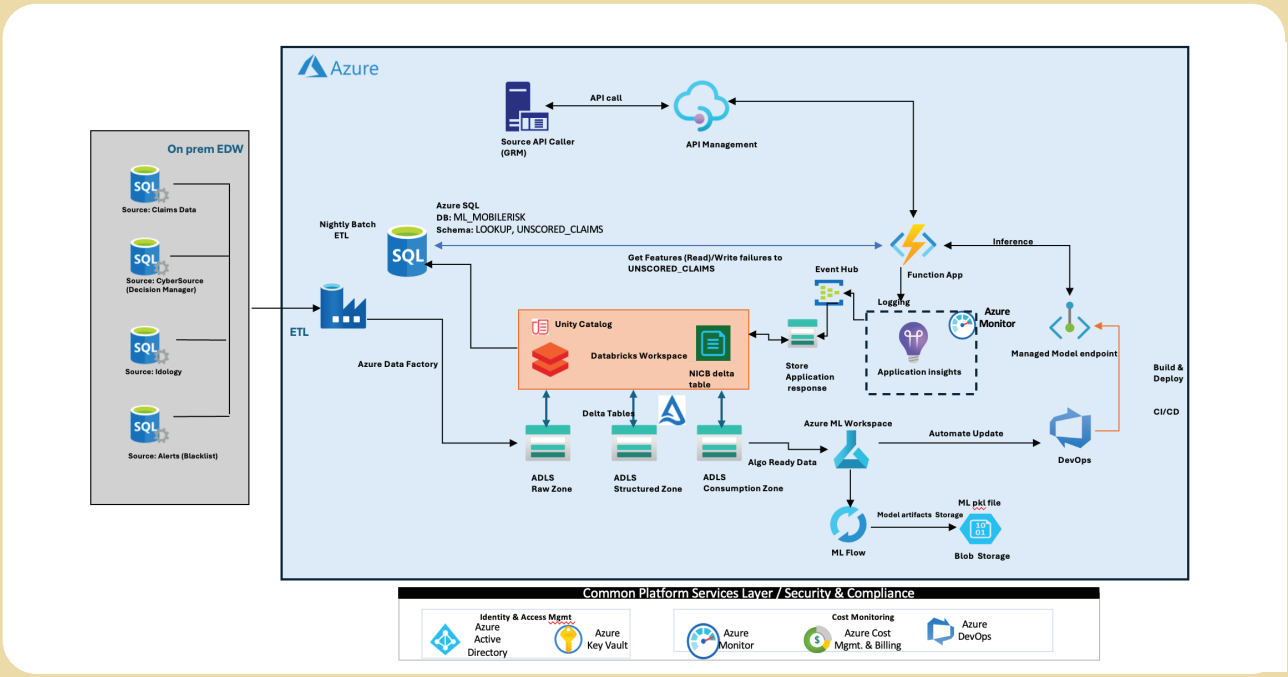


Figure 3: High-level solution architecture

The architecture, as depicted in Figure 3, presents a scalable, secure, and enterprise-grade Azure platform where data from multiple on-premise SQL systems, including claims, risk, identity, and alert sources, is ingested using Azure Data Factory and stored in Azure Data Lake Storage (ADLS) following a structured multi-zone design (raw, structured, and consumption layers). Azure Databricks, governed by Unity Catalog, performs data transformation, feature engineering, and Delta Lake management to produce an algorithm-ready dataset for model training and to load look-up tables for real-time inference. The inference data (look-up tables) will be refreshed daily and persisted in Azure SQL for operational consumption in real time. An ML or model training pipeline is created and managed in Azure Machine Learning, with ML flow for experiment tracking, model versioning, and artifact management, and model assets stored in Azure Blob Storage. CI/CD pipelines implemented using Azure DevOps automate model build, validation, and deployment to managed online endpoints. Real-time inference is enabled through Azure API Management and Azure Functions, which securely expose model predictions to the caller application (GRM). At the same time, Event Hub supports asynchronous logging to ADLS, which will be leveraged for model retraining pipelines and auditability. Comprehensive observability is achieved through Application Insights and Azure Monitor, and enterprise-wide security, compliance, and governance are enforced via Azure Active Directory, key vault, cost management, and centralized DevOps controls. Together, this architecture delivers an end-to-end, production-ready data and ML platform that supports real-time decisioning and enterprise operational excellence.

Data sources and volumes

The solution was built and validated using approximately 550,000 historical claims data spanning two years. The first seven quarters of data (Q1-Q7) were used for ML model training and validation, and the last quarter (Q8) was used for model testing. Fraud-risk intelligence is consolidated from five primary data sources to create a unified and comprehensive risk profile for each claim.
The **claimant history** dataset provides foundational information about claims, including device, enrollment, and address details. It is enriched with up to three years of historical behavioral data across mobile numbers, email addresses, customer accounts, shipping addresses, and serial numbers. **Account history** contributes to critical payment-risk signals such as AFS scores, AVS outcomes, and historical transaction behavior at the mobile and account level. **Identity verification** data captures verification attempts, failure patterns, and longitudinal identity risk indicators. **Alerts and watchlists** identify matches across claimant attributes, including email, mobile numbers, accounts, and fuzzy address patterns linked to previously suspicious activity. **NICB** data adds external fraud intelligence, including risky PII correlations and advanced fuzzy address risk scoring.

Together, these integrated data sources deliver a holistic view of identity, behavior, payment risk, and address-based fraud signals, enabling highly accurate, scalable, real-time fraud prediction.

Feature engineering

The final feature set integrates static device attributes (for example: mobile model name, manufacturer etc.), behavioral indicators, 3-year historical aggregates across multiple PII elements (mobile, email, account, address, serial number), real-time fuzzy match scores against NICB and alerts datasets, and Azure Function computed temporal features such as time-to-report (TTR) and time between claims (TBC). Together, the 104 engineered features provide a comprehensive representation of claimant identity consistency, historical behavior, transactional risk, and device-level signals, enabling our model to detect complex fraud patterns with high recall.

ML model development

The model development process, as shown in Figure 4 below, followed a structured, step-by-step training and evaluation workflow designed to balance predictive performance, interpretability, and real-time latency constraints. The full dataset was partitioned into a train-validation set and a hold-out test set using a time-based split, with the most recent quarter reserved exclusively for testing. This ensured that model performance generalized to future fraud patterns and closely reflected real-world deployment conditions.

As an initial baseline, three gradient-boosting algorithms, XGBoost, LightGBM, and CatBoost, were independently trained on the train-validation dataset using the same 104 engineered features derived from multisource behavioral, temporal, historical, and fuzzy-matching signals. Baseline model performance was evaluated and compared using key classification metrics, with the F1 score serving as the primary criterion for model selection due to the imbalanced nature of the fraud detection problem.

Following baseline evaluation, the best-performing model configuration was selected for further optimization. Hyperparameter tuning was performed using Optuna, leveraging the Tree-Structured Parzen Estimator (TPE) sampler and early-pruning to efficiently explore the parameter space while reducing computational cost. Cross-validation was incorporated during this phase to ensure model robustness and stability.

After identifying optimal hyperparameters, the selected model was retrained using the full train-validation dataset to maximize learning from historical data. The retrained model was then evaluated on the independent test dataset, and comprehensive performance metrics, including F1 score, precision, recall, and calibrated probability behavior, were generated. All experiments, parameters, metrics, thresholds, and artifacts were systematically logged using MLflow to support reproducibility, auditability, and governance.

To ensure reliable probability outputs for downstream decisioning, the model's raw prediction scores were passed through a beta calibration layer, producing calibrated probabilities suitable for threshold-based routing in production. The final calibrated model, along with feature importance and cross-validation results, was registered as the production-ready artifact, ensuring the solution is accurate, explainable, and capable of supporting sub-second real-time inference.

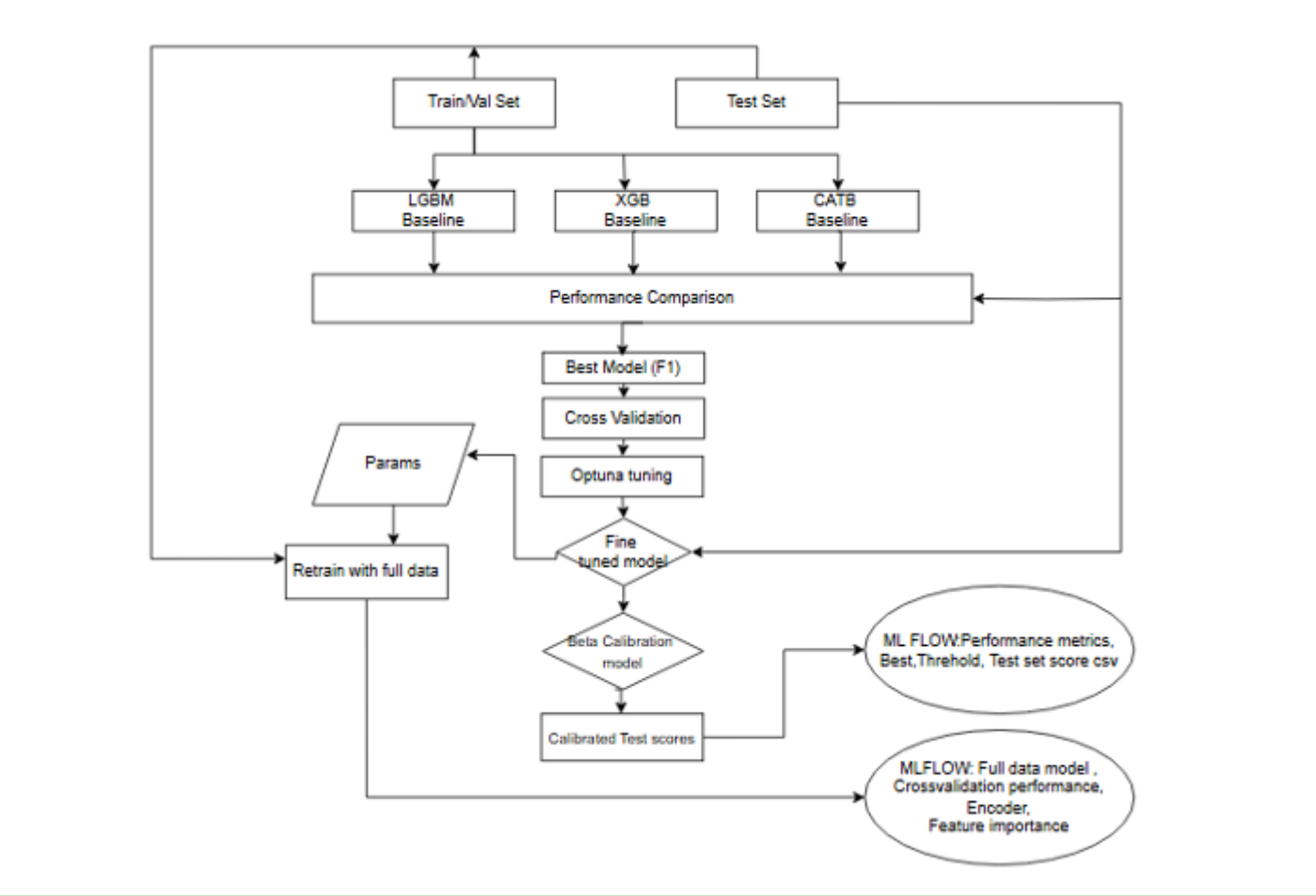


Figure 4: End-to-end model training and calibration workflow

Why Optuna was used?

Optuna[15] was selected as the hyperparameter optimization framework because it provides state-of-the-art automated search capabilities. It uses a technique called define-by-run that dynamically constructs search spaces, enabling more flexible, expressive optimization workflows than traditional libraries like GridSearchCV or RandomSearch. Optuna also employs a Bayesian-optimization-inspired sampler (TPE, Tree-structured Parzen Estimator), which learns from previous trials and prioritizes promising regions of the search space. This makes it more efficient and significantly faster than brute-force search. In this fraud detection use case, where feature interactions are complex and model performance is sensitive to hyperparameters, Optuna helped identify high-performing parameter combinations while minimizing computation time. Its pruning mechanism stopped poorly performing trials early, reducing cost and accelerating experimentation.

Model performance and business impact realized during user acceptance testing (UAT)

Following multiple rounds of experimentation across different algorithms, and with explainability as a primary requirement, the final model selection was a **tree-based XGBoost model**. This approach delivered balanced and reliable performance across key evaluation metrics, achieving an **F1 score of 0.56**, **precision of 0.54**, and **recall of 0.55**. These results reflect an effective trade-off between identifying fraudulent claims and minimizing false positives in a real-time scoring environment.

To assess real-world impact beyond quantitative metrics, false positives and false negatives were reviewed in collaboration with Claims Risk Reviewers using randomly sampled cases. The review revealed that **approximately 80%-85% of model-identified false positives represented genuinely risky or potentially fraudulent claims**, which had previously gone undetected under the existing process. This outcome significantly increased fraud visibility while maintaining operational confidence, representing a substantial improvement in risk capture rather than noise generation.

For the remaining cases not detected by the model, analysis confirmed that the key signals required for identification were **not present within assurant's internal data landscape**. These attributes were sourced from external systems routinely consulted

by investigators during manual review, validating the model's limitations as data-driven rather than algorithmic. This finding reinforced both the credibility of the model results and the broader data strategy, highlighting clear opportunities for future enrichment. Collectively, these outcomes demonstrate that the solution improves fraud detection effectiveness while remaining explainable, trustworthy, and aligned with operational realities.

Before implementation, analysis showed that approximately 75% of reviewer effort was spent reviewing legitimate ("good customer") claims, with the triage team manually reviewing 7,500-8,000 claims per month. This resulted in high operational overhead and limited capacity to focus on truly high-risk cases.

Based on UAT, the solution is expected to deliver meaningful efficiency and risk-reduction benefits. Monthly manual review volumes are projected to **decrease by approximately 40%, while the proportion of reviewer time spent on legitimate claims is expected to drop from 75% to 45%, significantly improving investigator productivity and focus.** In addition, the solution is **estimated to identify an additional 1,200 fraudulent or high-risk claims per month in real time**, cases that previously went undetected in the existing (as-is) GRM process. Together, these improvements drive lower operational costs, increased fraud capture, and faster, more confident decision-making for the business.

Explainability and model transparency

Explainability is a foundational component of the fraud detection architecture, ensuring that every model prediction can be clearly interpreted and justified by operational teams, fraud investigators, auditors, and compliance stakeholders. To meet this requirement, the solution leverages **SHAP (SHapley Additive exPlanations)**, a game-theoretic framework that decomposes each prediction into additive, feature-level contributions.

For tree-based models such as **XGBoost**, SHAP's **TreeExplainer** provides exact or near-exact feature-importance attribution with high computational efficiency. During early development, SHAP values were computed with approximate=True to support rapid experimentation. Through targeted optimizations within the Azure Function scoring pipeline, such as pre-loading SHAP background datasets and minimizing Python object serialization overhead, the system transitioned to approximate = False, delivering fully accurate SHAP explanations while consistently meeting sub-second latency service-level objectives.

Operationally, explainability supports three critical use cases. First, real-time, claim-level explanations are returned with each fraud score, including a ranked list of the most influential features. This allows investigators to quickly understand driving risk factors such as prior fraud history, NICB match indicators, abnormal reporting times, or address inconsistencies. Second, SHAP-based explanations support regulatory and audit compliance by providing mathematically sound, human-interpretable artifacts suitable for governance reviews, model documentation, and risk assessments. Third, SHAP values enable ongoing drift monitoring and model stability analysis by tracking changes in feature importance and behavioral patterns over time, allowing teams to identify emerging fraud trends and trigger retraining or recalibration proactively.

Together, these explainability capabilities ensure that the fraud detection system is not only accurate, but also transparent, accountable, and trusted, supporting effective operational decision-making and meeting enterprise-level risk and governance standards.

Operational challenges with model dependent thresholds

Threshold instability across models

During model experimentation, a critical operational challenge emerged: **threshold instability** across different machine-learning models. Three gradient-boosted tree models - XGBoost, CatBoost, and LightGBM were trained and evaluated independently using the same feature set, objective function, and evaluation metric (F1 score). Despite achieving comparable model performance, each model required a different optimal decision threshold, ranging from approximately **0.25 to 0.45**, to balance precision and recall.

At similar F1 score levels, XGBoost, CatBoost, and LightGBM each converged on distinct decision thresholds. When applied to the same test population, these model-specific thresholds resulted in **substantial variation in the volume of claims flagged for manual review**, even though the underlying fraud rate remained constant. In practice, switching models led to **double-digit percentage swings in flagged claim volumes,** directly affecting investigator workload, SLA adherence, and operational costs. Table 1 summarizes the above observations.

Model	Optimal threshold	% of suspicious claims flagged by the model	Daily review volume*	Comment
XGBoost	0.35	~7.0%	~700	Slightly above baseline*
CatBoost	0.25	~11.5%	~1,150	~2× baseline*
LightGBM	0.45	~4.5%	~450	Below baseline*

*Assuming approximately 10,000 claims scored per day. Baseline implies 6.2% of fraudulent claims detected by the GRM.

Table 1: Model thresholds vs daily claim review volume (triage team) without any calibration.

Business risk and need for threshold decoupling

The variability, as seen above, introduced a significant business risk. Our fraud detection solution was designed for periodic retraining using newly available data, creating the realistic possibility that a different model could emerge as the top performer in future refresh cycles. Without a mitigation, each model change would require:

- Redefining decision thresholds
- Re-aligning downstream fraud workflows
- Re-estimating investigation capacity
- Re-educating business users on score interpretation

Such frequent operational recalibration would undermine confidence in the system and make it difficult to maintain stable, predictable manual review volumes. The business, therefore, required a mechanism to decouple model selection from operational thresholds, ensuring that probability scores remained interpretable, stable, and comparable across models and retraining cycles.

Isotonic calibration: Exploration and limitations

As an intermediate step, Isotonic Calibration was evaluated to improve probability behavior while retaining model-specific thresholds. The goal was to reduce extreme over- or under-confidence in raw model outputs, rather than to enforce a single decision threshold.

Following isotonic calibration, optimal thresholds across models moved closer, falling within a narrower range (approximately 0.25 to 0.35). However, empirical analysis revealed a significant limitation. Because isotonic calibration applies to a piecewise-constant mapping, a large proportion of heterogeneous claims were assigned identical probability scores, even when their feature profiles and SHAP explanations differed materially. This score clustering weakened risk-based prioritization, forcing investigators to review claims with identical risk scores but different underlying fraud drivers. From an operational perspective, this flattened the review queue, reduced throughput, and increased review time per claim. From a governance standpoint, it introduced explainability ambiguity by assigning identical scores to claims with materially different SHAP explanations. Table 2 summarizes the above observations.

Model	Optimal threshold	% of suspicious claims flagged by the model	% claims sharing the same score	Business Impact
XGBoost	0.30	~6.8%	~25–30%	• Weak risk ranking
CatBoost	0.28	~7.2%	~30–40%	• Prioritization loss
LightGBM	0.33	~5.9%	~20–25%	• Explainability ambiguity

*Assuming approximately 10,000 claims scored per day.

Table 2: Model thresholds vs daily claim review volume (triage team) with isotonic calibration.

Beta calibration: Stabilizing probabilities and operations

To address the limitations discussed above, **Beta Calibration** was introduced as a post-model calibration layer. While gradient-boosted tree models excel at ranking fraud risk, their raw probability outputs are often poorly calibrated and overly confident, especially in highly imbalanced fraud datasets.

Beta calibration learns a flexible probability mapping using a Beta distribution, making it better suited than Platt scaling or isotonic regression for fraud detection use cases. The approach produces **smooth, well-calibrated probability scores** while adding negligible latency, making it suitable for real-time deployment.

Post beta calibration outcomes and business impact

After applying beta calibration, the probability distributions across XGBoost, CatBoost, and LightGBM aligned and stabilized, enabling the use of a consistent operational threshold regardless of the underlying model. This eliminated large swings in review volumes and simplified operational planning. By decoupling decision thresholds from model selection, beta calibration **enabled predictable operational volumes, consistent risk interpretation, and frictionless model refreshes**. This approach preserved model flexibility while delivering stable decisioning, improved investigator efficiency, and increased business confidence in the fraud detection platform. Table 3 summarizes the above observations.

Model	Beta calibrated threshold	% of suspicious claims flagged by the model	Daily review volume*	Outcome
XGBoost	~0.30	~6.2%	~620	Stable and aligned
CatBoost	~0.31	~6.4%	~640	
LightGBM	~0.29	~6.1%	~610	

*Assuming approximately 10,000 claims scored per day.

Table 3: Model thresholds vs daily claim review volume (triage team) with beta calibration.

Real-time inference

The real-time inference enables the underlying ML Model to secure, sub-second risk scoring by combining scalable cloud orchestration with consistent feature generation in the production environment. An Azure Function App serves as the central control plane, handling authentication, real-time feature enrichment, model invocation, and response orchestration. The system computes both instantaneous signals (on-the-fly calculated) and pre-computed risk features (historical features present in look-up tables) by performing low-latency look-ups against pre-indexed historical data in Azure SQL. This design ensures high performance, strong security, and feature parity while supporting real-time decisioning at scale.

Latency optimization

To meet sub-second end-to-end latency requirements, the architecture incorporates optimizations across compute, data access, and feature processing layers. Azure Function Apps are kept warm using always-on to eliminate cold-start delays, while frequently referenced static datasets and model metadata are cached in-memory to avoid repeated I/O. Historical features leverage pre-indexed tables and highly selective SQL queries for low-latency retrieval. Real-time feature computation is optimized through lightweight, vectorized Python operations, ensuring microsecond-level execution for time-based calculations, match indicators, and risk evaluations. Together, these optimizations enable consistent, high-performance real-time inference at scale.

Conclusion

The real-time fraud detection solution integrates multisource feature engineering, optimized XGBoost modeling, and cloud-native deployment to deliver fast, reliable, and scalable fraud risk scoring. A key differentiator is the use of Beta Calibration, which transforms raw model outputs into stable, actionable probabilities, enabling consistent decision thresholds and improving operational trust. By correcting the inherent overconfidence of gradient-boosted models, the solution ensures that elevated scores correspond to true business risk. Combined with a low-latency Azure Function inference pipeline and SHAP-based explainability, the platform delivers accurate, calibrated, and interpretable fraud predictions at enterprise scale.

References

[1] Coalition Against Insurance Fraud, "Fraud trends," 2026. [Online]. Available: https://insurancefraud.org/publications/fraud-trends/

[2] CSRAID, "Smartphone insurance fraud: Emerging threats in device protection programs," 2025. [Online]. Available: https://news.csraid.com/en/hub/phone-insurance-fraud-scam-protection-2025

[3] Didit, "Adaptive fraud scoring with Azure Functions," 2026. [Online]. Available: https://didit.me/blog/adaptive-fraud-scoring-azure-functions-didit/

[4] Equifax, "The shifting landscape of digital fraud: Why mobile is the new battleground," Nov. 5, 2025. [Online]. Available: https://www.equifax.com/business/blog/-/insight/article/the-shifting-landscape-of-digital-fraud-why-mobile-is-the-new-battleground/

[5] N. Gupta and S. Gupta, "ML-powered insurance fraud detection with explainability," Int. J. Sci. Adv. Res. Technol., 2025. [Online]. Available: https://www.ijsart.com/public/storage/paper/pdf/IJSARTV11I6103821.pdf

[6] J. Huang, "Insurance fraud identification based on XGBoost," Int. J. Applied Science and Mathematics, 2023. [Online]. Available: https://ijasm.org/administrator/components/com_jresearch/files/publications/IJASM_407_FINAL.pdf

[7] A. Khalil et al., "Enhancing insurance fraud detection accuracy with integrated machine learning and statistical methods," Computational Economics, 2025, doi: 10.1007/s10614-025-11074-0.

[8] M. Kull, T. M. Silva Filho, and P. Flach, "Beta calibration: A well-founded and easily implemented improvement on logistic calibration for binary classifiers," in Proc. AISTATS, 2017, pp. 623–631.

[9] E. Luzio et al., "Decoupling decision-making in fraud prevention through classifier calibration," in Proc. ACM Symposium on Applied Computing, 2024. [Online]. Available: https://arxiv.org/abs/2401.05240

[10] Microsoft, "Real-time fraud detection on Azure: Reference architecture," 2026. [Online]. Available: https://github.com/microsoft/azure-realtime-fraud-detection

[11] A. Narayanan, "Real-time fraud detection in cloud-native fintech systems," Global J. Eng. Technol. Adv., vol. 23, no. 1, pp. 410–419, 2025, doi: 10.30574/gjeta.2025.23.1.0087.

[12] TransUnion, "H2 2025 fraud trends report," 2025. [Online]. Available: https://media.transunion.com/content/dam/transunion/roa/business/collateral/report/3611950-h2-25-fraud-trends-rpr.pdf

[13] F. Vandervorst et al., "Inductive inference of gradient-boosted decision trees on graphs for insurance fraud detection," arXiv preprint, 2025. [Online]. Available: https://arxiv.org/abs/2510.05676

[14] Z. Wang et al., "A robust and interpretable ensemble machine learning model for predicting healthcare insurance fraud," Scientific Reports, vol. 15, no. 1, art. no. 218, 2025.

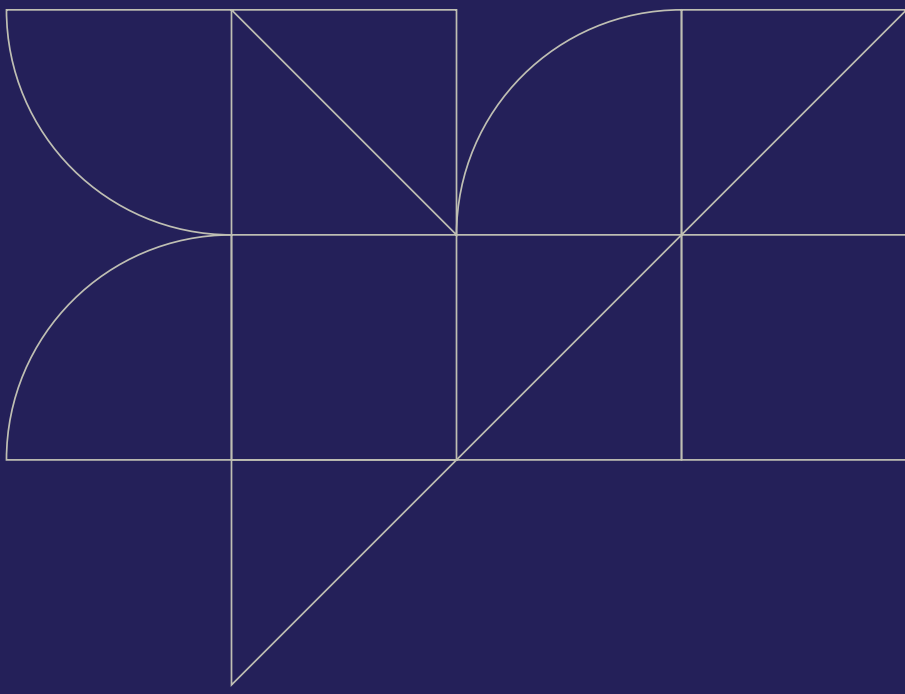
[15] https://optuna.org/

Authors:

Rahul Nimje,
Solution Architect
r.nimje@zensar.com

Anindya Chatterjee,
Solution Architect
anindya.chatterjee@zensar.com

Raghotham M R,
Engineering Manager
raghotham.mr@zensar.com



zensar
An RPG Company

At Zensar, we're 'experience-led everything.' We are committed to conceptualizing, designing, engineering, marketing, and managing digital solutions and experiences for over 145+ leading enterprises. Using our 3Es of experience, engineering, and engagement, we harness the power of technology, creativity, and insight to deliver impact.

Part of the $4.8 billion RPG Group, we are headquartered in Pune, India. Our 10,000+ employees work across 30+ locations worldwide, including Milpitas, Seattle, Princeton, Cape Town, London, Zurich, Singapore, and Mexico City.

For more information, please contact: **info@zensar.com | www.zensar.com**