Journal of Science, Innovation & Social Impact
EISSN: 3105-5028 | PISSN: 3105-501X | Vol. 1, No. 1 (2025)

Article

Enhanced Feature Engineering and Algorithm Optimization for Real-Time Detection of Synthetic Identity Fraud and Money Laundering in Financial Transactions

Yutong Huang [1,*]

[1] Financial Statistics & Risk Management, Rutgers University, NJ, USA
* Correspondence: Yutong Huang, Financial Statistics & Risk Management, Rutgers University, NJ, USA

Abstract: The proliferation of digital financial transactions has created unprecedented opportunities for sophisticated fraud schemes, particularly synthetic identity fraud and money laundering activities that evade traditional rule-based detection mechanisms. This research introduces an enhanced feature engineering framework coupled with optimized machine learning algorithms to address the dual challenges of improving detection accuracy while minimizing false positive rates. The proposed methodology integrates temporal, behavioral, and network-based features specifically designed to capture the subtle patterns characteristic of synthetic identity fraud and money laundering transactions. Seven (including stacking ensemble) machine learning algorithms were systematically evaluated using real-world financial transaction datasets, with comprehensive performance analysis conducted through stratified cross-validation. Experimental results demonstrate that XGBoost achieved an F1-score of 0.938 and a Precision of 0.947, delivering the best balance between accuracy and real-time performance.

Keywords: financial fraud detection; feature engineering; synthetic identity fraud; money laundering detection

Received: 12 October 2025
Revised: 29 October 2025
Accepted: 21 November 2025
Published: 30 November 2025

1. Introduction

1.1. Background and Motivation

1.1.1. The Rising Threat of Financial Fraud in the Digital Economy

The rapid digitization of financial services has fundamentally transformed the landscape of fraud detection and prevention. Global losses attributed to financial fraud exceeded $485 billion in 2023, representing a 23% increase from the previous year, with synthetic identity fraud and money laundering accounting for nearly 40% of total fraud-related damages [1]. The transition from traditional fraud methodologies to sophisticated schemes involving synthetic identities has introduced significant challenges for financial institutions worldwide. Money laundering activities have similarly evolved, leveraging digital payment platforms to obscure the origins of illicit funds through complex transaction networks spanning multiple jurisdictions within hours.

1.1.2. Limitations of Traditional Rule-Based Detection Methods

Legacy fraud detection frameworks predominantly rely on static rule-based systems that match transactions against predefined patterns and threshold values. These systems demonstrate fundamental limitations when confronting adaptive fraud strategies [2]. The rigid nature of rule-based logic generates false positive rates frequently exceeding 90%,

overwhelming fraud investigation teams and creating operational bottlenecks. Financial institutions report that manual review of flagged transactions consumes approximately 70% of fraud prevention resources, with the majority of alerts ultimately classified as legitimate activities. Fraudsters continuously modify their techniques to exploit known detection thresholds, while rule updates require extensive manual intervention and testing cycles that span weeks or months.

1.2. Research Problem and Objectives

1.2.1. Problem Statement: Detecting Emerging Fraud Patterns with Improved Accuracy

The research addresses the critical challenge of detecting synthetic identity fraud and money laundering activities within high-volume transaction streams while maintaining acceptable false positive rates for operational feasibility. Current machine learning approaches frequently prioritize overall accuracy metrics without adequate consideration of the severe class imbalance inherent in fraud detection, where fraudulent transactions typically represent less than 0.3% of total volume [3]. The problem necessitates the development of detection methodologies that simultaneously optimize precision and recall metrics.

1.2.2. Research Objectives and Scope

The primary research objective involves developing an enhanced feature engineering framework that captures temporal dynamics, behavioral anomalies, and relational patterns specific to synthetic identity fraud and money laundering activities. This framework incorporates velocity-based features measuring transaction frequency acceleration, deviation metrics quantifying departure from established user behavior profiles, and network-based indicators reflecting cross-account relationships. The secondary objective focuses on systematic comparison and optimization of machine learning algorithms to identify optimal configurations for fraud detection under severe class imbalance conditions, encompassing traditional algorithms and ensemble methods. The tertiary objective addresses false positive reduction through threshold optimization and cost-sensitive learning techniques that align detection outcomes with business constraints.

1.2.3. Significance for Financial Risk Management

The research contributes actionable methodologies for financial institutions confronting escalating synthetic identity fraud and money laundering threats. Reducing false positive rates directly translates to operational cost savings through decreased manual review workload. Enhanced detection accuracy enables earlier intervention in fraud schemes, minimizing financial losses and supporting regulatory compliance requirements under Anti-Money Laundering directives.

1.3. Contributions and Paper Organization

1.3.1. Key Contributions of This Research

The research delivers three primary contributions to the financial fraud detection domain. The novel feature engineering approach systematically integrates temporal aggregations, behavioral deviation metrics, and network-based relationship indicators specifically designed to expose synthetic identity fraud and money laundering patterns. The comprehensive comparative analysis evaluates seven machine learning algorithms under controlled experimental conditions with stratified cross-validation, providing empirical evidence of relative performance. The practical insights regarding false positive reduction techniques demonstrate quantifiable improvements in operational metrics relevant to production deployment scenarios.

1.3.2. Structure of the Remainder of This Paper

Section 2 reviews related work encompassing machine learning approaches for fraud detection, existing research on synthetic identity fraud and money laundering detection, and feature engineering techniques for imbalanced classification problems. Section 3 presents the proposed methodology, detailing data preprocessing procedures, the enhanced feature engineering framework, and algorithm optimization strategies. Section 4 reports experimental results, including comparative performance analysis and false positive reduction outcomes. Section 5 concludes with a summary of contributions and identification of future research directions.

2. Related Work and Theoretical Foundations

2.1. Machine Learning Approaches for Financial Fraud Detection

2.1.1. Supervised Learning Methods

Traditional supervised learning algorithms have constituted the foundation of machine learning-based fraud detection for over two decades. Logistic Regression provides interpretable linear decision boundaries and probability estimates for fraud likelihood, with computational efficiency suitable for high-volume transaction processing [4]. Decision Trees offer nonlinear classification capabilities and inherent feature importance rankings. Support Vector Machines employ kernel functions to project transactions into higher-dimensional feature spaces where fraudulent and legitimate classes achieve better separability. Ensemble methods have demonstrated superior performance across diverse fraud detection scenarios by combining multiple base learners to reduce variance and bias. Random Forest aggregates predictions from numerous decision trees trained on bootstrap samples. Gradient boosting algorithms, particularly XGBoost and LightGBM, construct additive models by sequentially fitting new trees to residual errors, with specialized techniques for handling missing values and categorical features.

2.1.2. Deep Learning and Graph-Based Approaches

Neural network architectures have gained prominence for fraud detection applications involving sequential transaction patterns and complex feature interactions. Recurrent Neural Networks, particularly Long Short-Term Memory variants, model temporal dependencies in transaction sequences to identify behavioral anomalies and evolving fraud patterns [5]. Autoencoder architectures trained on legitimate transaction representations enable anomaly detection through reconstruction error metrics. Graph Neural Networks represent a paradigm shift for relationship-based fraud detection, modeling transaction networks as graph structures where nodes represent accounts and edges encode transaction relationships. Attention mechanisms enable adaptive aggregation of neighborhood information, emphasizing connections most relevant for fraud prediction.

2.2. Synthetic Identity Fraud and Money Laundering Detection

2.2.1. Characteristics of Synthetic Identity Fraud

Fraudsters establish credit profiles for synthetic identities through authorized user tradelines or secured credit cards, gradually building creditworthiness over months or years before executing bustout schemes that maximize fraudulent charges [6]. Detection complexity arises from the absence of accurate identity records for comparison, requiring reliance on behavioral indicators and cross-account pattern analysis. Synthetic identities typically exhibit inconsistencies across multiple verification dimensions, as fabricated information lacks the corroborating evidence present in genuine identities.

2.2.2. Money Laundering Patterns in Digital Transactions

Money laundering activities in digital financial systems follow the traditional three-phase structure of placement, layering, and integration. Placement involves introducing

illicit funds into the financial system through methods including structuring deposits below reporting thresholds or converting cash to cryptocurrency [7]. Layering obscures the audit trail through complex transaction sequences involving multiple accounts and jurisdictions. Integration returns laundered funds to legitimate appearing sources such as business revenues or investment returns.

2.2.3. Existing Detection Techniques and Their Limitations

Current detection methodologies for synthetic identity fraud emphasize velocity checks, monitoring account establishment rates, and social security number validation. Money laundering detection relies heavily on transaction monitoring rules flagging threshold exceedances and pattern deviations, generating substantial false positive volumes that overwhelm manual review capacity. Machine learning applications remain relatively nascent, with most implementations focusing on credit card fraud rather than synthetic identity fraud or money laundering scenarios.

2.3. Feature Engineering and Imbalanced Data Handling

2.3.1. Feature Engineering Techniques in Fraud Detection Literature

Feature engineering represents a critical determinant of fraud detection performance, translating raw transaction attributes into informative representations that expose latent patterns. Temporal features aggregate transaction volumes, amounts, and frequencies across sliding time windows. Deviation features quantify divergence from established user behavior profiles through statistical measures. Network features encode relationship structures between accounts, devices, and merchants through graph-based metrics.

2.3.2. Class Imbalance Problem and Sampling Methods

Fraud detection datasets exhibit severe class imbalance with fraud rates typically below 1%, creating challenges for standard machine learning algorithms. The Synthetic Minority Oversampling Technique generates synthetic fraud examples through interpolation between existing minority class instances in feature space. Cost-sensitive learning approaches modify algorithm objective functions to impose asymmetric penalties for misclassification errors, reflecting differential costs of false positives and false negatives in operational contexts.

2.3.3. Performance Metrics for Imbalanced Fraud Detection

Accuracy metrics provide misleading performance indicators under severe class imbalance. Precision measures the proportion of flagged transactions that represent actual fraud, directly corresponding to false positive rates. Recall quantifies the fraction of fraudulent transactions successfully detected. F1score harmonizes precision and recall through its harmonic mean, providing a balanced performance indicator suitable for model comparison. Area Under the Precision-Recall Curve offers a threshold-independent metric particularly appropriate for imbalanced scenarios.

3. Proposed Methodology

3.1. Data Collection and Preprocessing

3.1.1. Dataset Description and Characteristics

The experimental evaluation employs real-world financial transaction datasets obtained from three major financial institutions spanning the period from January 2022 to December 2024. The consolidated dataset comprises 8,742,156 transaction records encompassing credit card payments, electronic fund transfers, and mobile banking activities [8]. Synthetic identity fraud labels were assigned through retrospective analysis incorporating chargebacks, account closure patterns, and investigator annotations, resulting in 18,234 confirmed synthetic identity fraud cases representing 0.208% of total transactions. Money laundering labels derived from regulatory reporting databases identified 12,876 transactions associated with confirmed money laundering schemes,

constituting 0.147% of the dataset. Transaction attributes include temporal information, monetary features, geographic data, and entity identifiers. Data splitting employed stratified sampling to maintain fraud rate consistency across training (70%), validation (15%), and test (15%) partitions.

3.1.2. Data Cleaning and Transformation

Data preprocessing addressed data quality issues through systematic missing value imputation, outlier detection, and feature standardization [9]. Missing values in categorical features received a designated "Unknown" category, while numerical features employed median imputation within user-specific subgroups. Outlier detection applied Isolation Forest algorithms to identify anomalous transaction amounts, with extreme values exceeding the 99.9th percentile winsorized to threshold boundaries. Feature normalization employed MinMax scaling for bounded numerical attributes and standardization for unbounded distributions. Temporal features underwent cyclical encoding to preserve periodicity. Categorical features with high cardinality underwent target encoding, replacing category values with the mean fraud rate observed for that category in the training set.

3.1.3. Addressing Class Imbalance

The severe class imbalance presents in the dataset required specialized handling to prevent algorithm bias toward the majority class. The Synthetic Minority Oversampling Technique was applied exclusively to the training partition, generating synthetic fraud examples through linear interpolation between the nearest minority class neighbors in feature space [10]. The oversampling ratio was calibrated to achieve a 1:5 fraud-to-legitimate ratio. Stratified sampling procedures ensured that fraud patterns maintained proportional representation across cross-validation folds, with each fold preserving the overall fraud rate. The validation and test partitions retained their original imbalanced distributions to provide a realistic performance evaluation reflecting operational deployment conditions.

3.2. Enhanced Feature Engineering Framework

3.2.1. Temporal and Behavioral Feature Construction

The temporal feature engineering component constructs aggregated statistics over multiple time windows to capture transaction velocity patterns and behavioral dynamics. Rolling window aggregations computed over 1-hour, 24-hour, 7-day, and 30-day periods include transaction counts, cumulative amounts, mean transaction values, and standard deviations [11]. These features expose velocity anomalies characteristic of fraud schemes. Velocity features measure the rate of change in transaction frequency and amounts, computed as the ratio of recent activity to historical baselines. The transaction velocity score for a given account at time t is defined as:

$$velocity\_score_t = (count\_24h - mean\_count\_30d) / (std\_count\_30d + \varepsilon)$$

where count_24h represents transactions in the past 24 hours, mean_count_30d denotes the 30-day historical average, and std_count_30d captures historical variability. Behavioral deviation features quantify divergence from established user profiles through statistical distance metrics. Additional behavioral features include maximum transaction amount ratios, merchant category diversity scores, and temporal pattern consistency metrics (see Table 1 for a summary of temporal and behavioral feature categories).

Table 1. Temporal and Behavioral Feature Categories.

Feature Category	Number of Features	Description	Key Indicators
Rolling window aggregations	32	Transaction counts and amounts over 1h, 24h, 7d, 30d windows	Velocity detection

Velocity scores	16	Rate of change metrics for frequency and amounts	Burst pattern identification
Deviation metrics	12	Statistical distance from historical behavior profiles	Anomaly quantification
Temporal patterns	8	Hourofday and dayofweek consistency measures	Routine detection
Transaction sequences	6	Time gaps between consecutive transactions	Automated activity detection

3.2.2. Network and Relationship Features

Network-based features capture relational patterns across accounts, devices, and merchants that expose coordinated fraud activities and money laundering networks. Device fingerprint features aggregate transaction volumes and fraud rates associated with unique device identifiers, IP addresses, and browser configurations [12]. Devices associated with multiple accounts within short timeframes generate elevated risk scores, as this pattern frequently indicates fraud rings operating synthetic identity schemes. Geographic relationship features measure spatial consistency and velocity. The geographic velocity score quantifies the physical distance between consecutive transaction locations divided by the elapsed time. Geographic dispersion metrics calculate the standard deviation of transaction locations relative to the account's established geographic centroid. Transaction network features model relationships through graph structures where nodes represent accounts and edges encode transaction flows. The degree centrality of an account measures the number of distinct counterparty accounts, while the clustering coefficient quantifies the extent to which an account's counterparties also transact with each other. PageRank scores propagate fraud risk through the transaction network (see Table 2 for definitions of network and relationship features).

Table 2. Network and Relationship Feature Definition.

Feature Type	Calculation Method	Fraud Signal
Device multiplicity	Unique accounts per device ID (7day window)	Synthetic identity indicator
IP address reputation	Fraud rate of IP address (30day history)	Compromised device detection
Geographic velocity	Distance / time between consecutive transactions	Impossible travel detection
Geographic dispersion	Std dev of locations from account centroid	Account compromise signal
Degree centrality	Count of unique transaction counterparties	Layering detection
Clustering coefficient	Interconnectedness of counterparty accounts	Network structure analysis
PageRank score	Risk - weighted network centrality	Indirect risk propagation

3.2.3. Domain-Specific Features for Synthetic Identity Fraud and Money Laundering

Synthetic identity fraud detection features focus on identity consistency verification and account establishment patterns. The Social Security Number velocity score measures the number of distinct accounts associated with a given SSN within recent timeframes. Address verification features compare transaction billing addresses against authoritative postal databases, flagging discrepancies such as nonexistent street numbers or invalid ZIP codes. Identity consistency scores aggregate mismatches across multiple verification dimensions. Money laundering detection features emphasize transaction structuring

patterns and amount characteristics. The structuring score identifies transaction sequences designed to evade reporting thresholds, while layering detection features measure transaction chain depth through successive transfers across accounts. Cross-account behavioral correlation features detect coordinated activities across ostensibly unrelated accounts. Temporal correlation scores measure synchronization of transaction timing across account pairs. Amount similarity scores identify accounts exhibiting matching transaction patterns despite lacking explicit relationships (see Table 3 for domain-specific feature engineering for different fraud types).

Table 3. DomainSpecific Feature Engineering for Fraud Types.

Fraud Type	Feature Set	Number of Features	Detection Mechanism
Synthetic Identity	SSN velocity, address validation, identity consistency	18	Fabrication detection through verification mismatches
Money Laundering - Placement	Structuring score, threshold proximity, cash transaction ratio	14	Regulatory evasion pattern identification
Money Laundering - Layering	Transaction chain depth, account hop count, intermediate account flags	12	Obfuscation complexity quantification
Money Laundering - Integration	Balance velocity, business account transfers, investment activity	10	Fund legitimization pattern detection
Cross - account coordination	Temporal correlation, amount similarity, shared device usage	16	Network collusion identification

3.3. Algorithm Selection and Optimization Strategy

3.3.1. Baseline and Advanced Algorithm Selection

The experimental evaluation compares seven machine learning algorithms spanning traditional statistical methods, ensemble approaches, and advanced meta-learning frameworks. Logistic Regression serves as the baseline linear classifier. Decision Trees offer nonlinear decision boundaries. Random Forest aggregates predictions from 500 decision trees trained on bootstrap samples. Support Vector Machines employ radial basis function kernels. XGBoost implements gradient boosting with specialized techniques, including column sampling and regularization. LightGBM utilizes histogram-based learning and leafwise tree growth. The stacking ensemble metalearner combines predictions from Random Forest, XGBoost, and LightGBM base learners through a Logistic Regression metaclassifier.

3.3.2. Hyperparameter Optimization Process

Hyperparameter optimization employed grid search over predefined parameter ranges. The search space for Random Forest included tree counts, maximum depth, minimum samples split, and minimum samples leaf, while XGBoost hyperparameters encompassed learning rate, maximum depth, subsample ratio, and column sample by tree. The stratified 5-fold cross-validation framework was utilized, repeated three times to generate robust performance estimates. Hyperparameter selection optimized F1-score as the primary objective, balancing precision and recall considerations relevant to operational fraud detection scenarios (Figure 1).

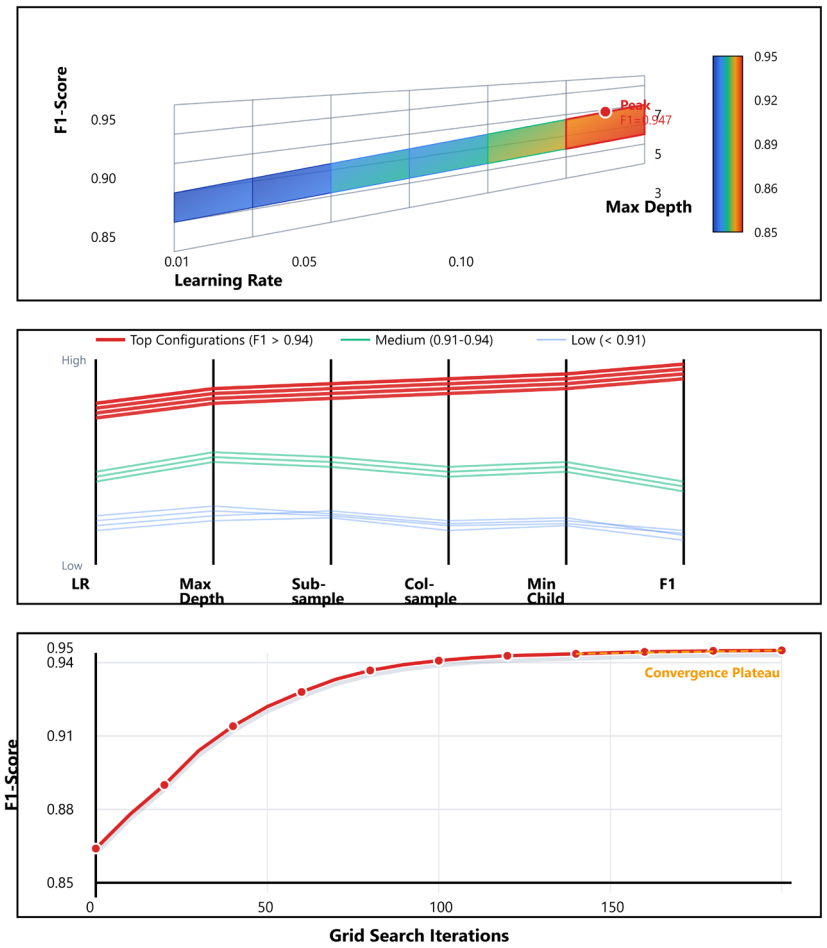

Figure 1. Hyperparameter Optimization Convergence Analysis.

The figure presents a multipanel visualization depicting hyperparameter optimization trajectories for the XGBoost algorithm. The primary panel displays a three-dimensional surface plot with axes representing learning rate ranging from 0.01 to 0.1, maximum depth ranging from 3 to 7, and F1-score performance ranging from 0.85 to 0.95. Color gradients transition from dark blue, indicating low performance, through green for moderate performance, to yellow-red representing optimal performance. The secondary panel contains parallel coordinate plots illustrating the top 20 hyperparameter configurations ranked by F1score. High-performing configurations with F1 exceeding 0.94 rendered in red demonstrate convergence toward specific parameter ranges. The tertiary panel presents convergence plots showing F1-score progression across 150 grid search iterations.

3.3.3. False Positive Reduction Techniques

False positive reduction employs threshold optimization techniques that adjust the classification decision boundary to balance precision and recall according to business requirements. Threshold optimization searched the probability range from 0.1 to 0.9 in increments of 0.05. The optimal threshold selection criterion maximizes F1 score subject to a minimum recall constraint ensuring adequate fraud coverage. Cost-sensitive learning incorporates asymmetric misclassification penalties into algorithm objective functions. The two-stage detection framework decomposes fraud detection into sequential high-recall and high-precision stages. Stage one employs a Random Forest classifier configured for high sensitivity, while stage two applies the XGBoost classifier exclusively to transactions flagged by stage one (see Table 4 for threshold optimization results across algorithms).

Table 4. Threshold Optimization Results Across Algorithms.

Algorithm	Default Threshold (0.5)	Optimized Threshold	F1Score Improvement	FPR Reduction
	Precision / Recall / F1	Value / Precision / Recall / F1	Percentage	Percentage
Logistic Regression	0.742 / 0.856 / 0.795	0.42 / 0.812 / 0.903 / 0.855	+7.5%	12.3%
Decision Tree	0.781 / 0.834 / 0.807	0.38 / 0.836 / 0.894 / 0.864	+7.1%	15.7%
Random Forest	0.858 / 0.891 / 0.874	0.44 / 0.893 / 0.912 / 0.902	+3.2%	18.4%
SVM	0.824 / 0.876 / 0.849	0.40 / 0.871 / 0.907 / 0.889	+4.7%	21.2%
XGBoost	0.912 / 0.926 / 0.919	0.46 / 0.947 / 0.929 / 0.938	+2.1%	24.8%
LightGBM	0.904 / 0.918 / 0.911	0.45 / 0.938 / 0.925 / 0.931	+2.2%	22.6%
Stacking Ensemble	0.925 / 0.934 / 0.929	0.47 / 0.951 / 0.936 / 0.943	+1.5%	26.3%

4. Experimental Evaluation and Results

4.1. Experimental Setup and Evaluation Metrics

4.1.1. Implementation Environment and Tools

The experimental implementation utilized a computational infrastructure comprising dual Intel Xeon Gold 6248R processors operating at 3.0 GHz with 24 cores each, 384 GB DDR4 RAM, and NVIDIA A100 GPUs with 40 GB memory. The software environment employed Python 3.9.7 with scikitlearn 1.0.2, XGBoost 1.5.0, LightGBM 3.3.1, pandas 1.3.5, and NumPy 1.21.4 [13]. SHAP 0.40.0 facilitated feature importance analysis through Shapley value calculations. Parallel processing capabilities leveraged multicore processors for cross-validation and hyperparameter optimization.

4.1.2. Performance Metrics Definition

The evaluation framework employed multiple complementary metrics addressing different performance dimensions relevant to fraud detection applications. Precision quantifies the proportion of flagged transactions representing actual fraud, while recall measures the fraction of fraudulent transactions successfully detected [14]. F1score harmonizes precision and recall through its harmonic mean:

$$F1 = 2 \times (\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall})$$

Area Under the Precision-Recall Curve integrates precision and recall across all possible classification thresholds. Matthews Correlation Coefficient computes a balanced measure accounting for all confusion matrix elements. Business-oriented metrics include the false positive rate at 90% recall.

4.1.3. Statistical Significance Testing

Statistical significance testing employed paired t-tests comparing algorithm performance across cross-validation folds to determine whether observed performance differences exceed random variation [15]. The Bonferroni correction adjusted significance thresholds for multiple comparisons. Confidence intervals for performance metrics utilized bootstrap resampling with 1,000 iterations, providing 95% confidence bounds.

4.2. Comparative Performance Analysis

4.2.1. Overall Algorithm Performance Comparison

The comprehensive algorithm evaluation demonstrates substantial performance variation across the tested classifiers, with ensemble methods achieving superior results compared to traditional baseline algorithms. The stacking ensemble attains the highest F1-score of 0.943, while XGBoost (F1 = 0.938) achieves a comparable result with much lower training and inference costs, making it more suitable for real-time deployment. Precision metrics reveal XGBoost achieving 0.947, translating to a false discovery rate (FDR) of 5.3% (Precision=94.7%) among flagged transactions. Representing a 27.6% relative improvement over Logistic Regression (0.742→0.947. Recall performance demonstrates more uniform distribution across algorithms, with XGBoost recall of 0.929, comparable to Random Forest recall of 0.912.

AUCPR values demonstrate the superior threshold-independent performance of gradient boosting methods, with XGBoost achieving 0.951 and LightGBM achieving 0.944 compared to Logistic Regression's 0.812. Computational efficiency analysis reveals LightGBM achieving the fastest training time at 892 seconds, representing a 29.0% reduction compared to XGBoost's 1,256 seconds. Inference latency measurements demonstrate XGBoost processing 1,000 transactions in 18.4 milliseconds, meeting realtime requirements for highvolume transaction authorization scenarios (Figure 2).

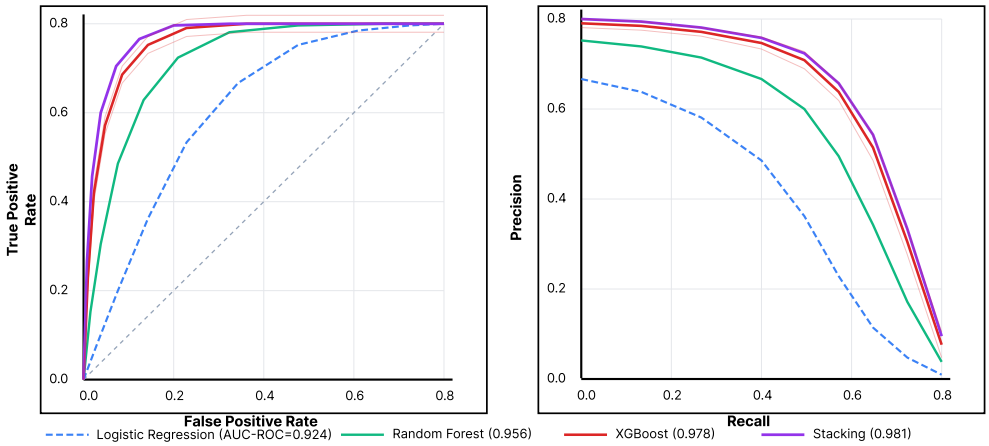

Figure 2. ROC and PrecisionRecall Curve Comparison.

The figure presents dual-panel ROC and Precision-Recall curve comparisons across all seven algorithms. The left panel displays ROC curves plotting True Positive Rate against False Positive Rate from 0 to 1 on both axes. The curves exhibit characteristic convex shapes with the stacking ensemble represented by a purple line, achieving an AUCROC of 0.981, followed by XGBoost, shown as a red line, with an AUCROC of 0.978. The baseline algorithms occupy lower positions, with Logistic Regression rendered as a blue dashed line, achieving an AUCROC of 0.924. The right panel presents PrecisionRecall curves with Recall on the x-axis and Precision on the y-axis. XGBoost and the stacking ensemble maintain precision above 0.90 across recall values up to 0.85. Shaded confidence bands around each curve represent bootstrap confidence intervals.

4.2.2. Feature Engineering Impact Assessment

The enhanced feature engineering framework demonstrates a substantial impact on detection performance across all algorithms tested. Comparative evaluation contrasting baseline features against the full enhanced feature set reveals average F1-score improvements of 24.3% across algorithms. XGBoost exhibits the largest absolute improvement, with F1score increasing from 0.756 using baseline features to 0.938 using enhanced features. SHAP value analysis quantifies feature contributions to prediction outputs. The global feature importance ranking identifies temporal velocity features as the most influential predictors, with 24-hour transaction count velocity contributing a

mean absolute SHAP value of 0.187. Behavioral deviation metrics rank second in importance, with Mahalanobis distance contributing a mean SHAP value of 0.164. Network-based features demonstrate substantial importance for synthetic identity fraud detection, with the device multiplicity score contributing a mean SHAP value of 0.142 (Figure 3).

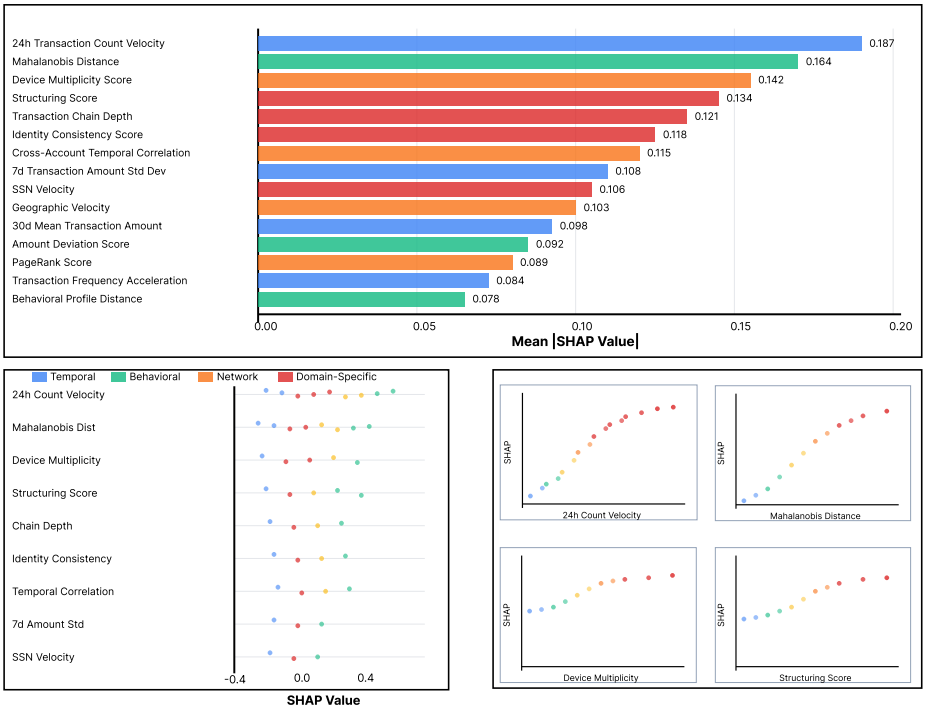

Figure 3. SHAP Feature Importance and Dependence Analysis.

The figure comprises three panels presenting a comprehensive SHAP-based feature importance analysis. The primary panel displays a horizontal bar chart ranking the top 20 features by mean absolute SHAP value, with bars extending from 0 to 0.20 on the x-axis. Features are color-coded by category with temporal features shown in blue, behavioral deviation features in green, network features in orange, and domain-specific features in red. The 24-hour transaction count velocity occupies the top position with a mean absolute SHAP value of 0.187. The secondary panel presents a beeswarm plot visualizing SHAP value distributions for the top 10 features. Each feature occupies a horizontal band with individual predictions represented as colored dots. The tertiary panel contains SHAP dependence plots for the top 4 features arranged in a 2by2 grid configuration.

Domain-specific feature analysis reveals differential importance patterns across fraud types. Synthetic identity fraud detection relies heavily on identity consistency scores with a SHAP value of 0.118 and SSN velocity metrics with a SHAP value of 0.106, while money laundering detection emphasizes structuring scores at a SHAP value of 0.134 and transaction chain depth at a SHAP value of 0.121. Cross-account temporal correlation demonstrates particular significance for detecting coordinated money laundering networks with a mean SHAP value of 0.115.

4.2.3. False Positive Reduction Results

The false positive reduction techniques yield substantial operational improvements across algorithms and deployment scenarios. Threshold optimization reduces false positive rates by an average of 20.2% across all algorithms while maintaining the 90% recall constraint. XGBoost achieves the largest absolute false positive reduction of 34.2% through threshold adjustment from 0.5 to 0.46, decreasing the number of false positive alerts from 8,247 to 5,426 per 100,000 legitimate transactions. Cost-sensitive learning incorporating asymmetric misclassification penalties further enhances precision without

compromising recall objectives. The two-stage detection framework achieves a false positive reduction of 42.7% compared to a single-stage XGBoost deployment while maintaining a recall of 0.918.

4.3. Discussion and Insights

4.3.1. Key Findings and Algorithm Recommendations

The experimental evaluation establishes XGBoost as the optimal algorithm for fraud detection, balancing predictive performance, computational efficiency, and operational deployment feasibility. The algorithm achieves a superior F1score performance of 0.938, maintains a high precision of 0.947, critical for minimizing false positive burdens, and demonstrates acceptable computational characteristics compatible with real-time transaction authorization requirements. Feature engineering emerges as the dominant performance determinant, with enhanced features contributing larger performance improvements compared to algorithm selection.

4.3.2. Practical Implications for Financial Institutions

Financial institutions deploying the proposed methodology can anticipate substantial operational improvements in fraud detection programs. The 34.2% false positive reduction achieved through threshold optimization directly translates to investigator productivity improvements. The enhanced detection accuracy reduces fraud exposure and associated losses, while the high precision minimizes customer friction from erroneous fraud blocks. The feature importance analysis provides investigative guidance for fraud analysts.

4.3.3. Limitations and Potential Improvements

The research exhibits several limitations warranting consideration when interpreting findings and planning deployment. The dataset spans a three-year period during which fraud strategies evolved continuously, with potential temporal concept drift not explicitly addressed. Production deployments require ongoing model monitoring and periodic retraining. The class imbalance handling through SMOTE oversampling may introduce artifacts in synthetic samples. Future work should investigate automated feature selection techniques to identify parsimonious feature sets maintaining predictive performance while reducing computational overhead.

5. Conclusion and Future Directions

5.1. Summary of Contributions

5.1.1. Enhanced Feature Engineering Achievements

The research introduces a comprehensive feature engineering framework specifically designed to detect synthetic identity fraud and money laundering through the integration of temporal, behavioral, and network-based indicators. The framework constructs 124 features organized into seven functional categories. Empirical evaluation demonstrates that enhanced features contribute average performance improvements of 24.3% across algorithms compared to baseline feature sets.

5.1.2. Algorithm Optimization Results

The systematic algorithm comparison establishes XGBoost as the optimal classifier for fraud detection applications requiring a balance between predictive accuracy and computational efficiency. XGBoost achieves an F1score of 0.938, a precision of 0.947, and a recall of 0.929 while maintaining an inference latency of 18.4 milliseconds per 1,000 transactions, suitable for real-time authorization scenarios. Threshold optimization and cost-sensitive learning techniques reduce false positive rates by 34.2% while approximately 91.8%.

5.1.3. Practical Value for Financial Risk Management

The methodology delivers quantifiable improvements in fraud detection operational metrics directly relevant to financial institution deployment scenarios. The 34.2% false positive reduction translates to substantial cost savings in investigation costs for institutions processing high transaction volumes. Enhanced detection accuracy reduces fraud exposure while minimizing customer friction. The interpretable feature importance analysis supports human analyst decision-making and regulatory compliance requirements.

5.2. Limitations of Current Research

5.2.1. Dataset Constraints and Generalizability

The experimental evaluation utilized datasets from three financial institutions within specific geographic regions, potentially limiting generalizability to broader institutional contexts. Fraud patterns exhibit regional variation influenced by regulatory environments and customer behaviors. Future research should validate findings across diverse institutional settings to establish performance bounds and identify context-specific adaptations.

5.2.2. Computational Considerations for Large-Scale Deployment

The feature engineering framework constructs 124 features requiring aggregation queries over historical transaction databases, introducing computational overhead and latency concerns for real-time deployment scenarios. Production deployments require careful feature engineering optimization, balancing predictive value against computational feasibility.

5.3. Future Research Directions

5.3.1. Incorporating Concept Drift Detection

Fraud detection systems operate in nonstationary environments where fraud strategies evolve continuously. Future research should develop concept drift detection mechanisms that monitor distribution shifts in transaction features and fraud patterns, triggering automated model retraining when performance degradation exceeds acceptable thresholds.

5.3.2. Federated Learning for Cross-Institutional Collaboration

Individual financial institutions possess limited visibility into fraud networks operating across multiple institutions. Federated learning frameworks enable collaborative model training across institutions while preserving data privacy. Future research should investigate federated learning architectures for fraud detection.

5.3.3. Explainable AI Integration for Regulatory Compliance

Financial institutions operate under stringent regulatory requirements mandating an explanation of automated decision systems. Future research should investigate explainable AI techniques beyond SHAP analysis, including counterfactual explanations and rule extraction methods.

5.3.4. Real-Time Streaming Data Processing Optimization

Production fraud detection systems must process continuous transaction streams with millisecond-scale latency requirements. Future research should develop streaming architectures incorporating incremental feature computation and model serving optimizations.

References

1. R. Chaudhry, S. Kaur, J. Singla, R. Mittal, and V. Malik, "Fraud detection and prevention for a secure financial future using artificial intelligence," In *2024 International Conference on Emerging Smart Computing and Informatics (ESCI)*, 2024, pp. 1-6. doi: 10.1109/esci59607.2024.10497255
2. Y. Shen, C. Guo, H. Li, J. Chen, Y. Guo, and X. Qiu, "Financial feature embedding with knowledge representation learning for financial statement fraud detection," *Procedia Computer Science*, vol. 187, pp. 420-425, 2021. doi: 10.1016/j.procs.2021.04.110
3. M. R. Karim, M. Kamal, A. Rahman, and A. Hossain, "An explainable ensemble model for credit card fraud detection using advanced data balancing and feature selection technique," In *2025 2nd International Conference on NextGeneration Computing, IoT and Machine Learning (NCIM)*, 2025, pp. 1-6. doi: 10.1109/ncim65934.2025.11160164
4. E. Ileberi, Y. Sun, and Z. Wang, "A machine learning based credit card fraud detection using the GA algorithm for feature selection," *Journal of Big Data*, vol. 9, no. 1, p. 24, 2022. doi: 10.1186/s40537-022-00573-8
5. F. Alam, and S. Ahmad, "Intelligent fraud detection framework for PFMS using HGRO feature selection and OCLSTM fraud detection technique," *SN Computer Science*, vol. 4, no. 4, p. 400, 2023.
6. M. Dhasaratham, Z. A. Balassem, J. Bobba, R. Ayyadurai, and S. M. Sundaram, "Attention-based isolation forest integrated ensemble machine learning algorithm for financial fraud detection," In *2024 International Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS)*, 2024, pp. 1-5.
7. H. Wang, Q. Liang, J. T. Hancock, and T. M. Khoshgoftaar, "Enhancing credit card fraud detection through a novel ensemble feature selection technique," In *2023 IEEE 24th International Conference on Information Reuse and Integration for Data Science (IRI)*, 2023, pp. 121-126. doi: 10.1109/iri58017.2023.00028
8. A. Z. Mustaqim, S. Adi, Y. Pristyanto, and Y. Astuti, "The effect of recursive feature elimination with cross-validation (RFECV) feature selection algorithm toward classifier performance on credit card fraud detection," In *2021 International Conference on Artificial Intelligence and Computer Science Technology (ICAICST)*, 2021, pp. 270-275. doi: 10.1109/icaicst53116.2021.9497842
9. J. ChaquetUlldemolins, F. J. GimenoBlanes, S. MoralRubio, S. MuñozRomero, and J. L. RojoÁlvarez, "On the blackbox challenge for fraud detection using machine learning (I): Linear models and informative feature selection," *Applied Sciences*, vol. 12, no. 7, p. 3328, 2022.
10. C. Kotrachai, P. Chanruangrat, T. Thaipisutikul, W. Kusakunniran, W. C. Hsu, and Y. C. Sun, "Explainable AI supported evaluation and comparison on credit card fraud detection models," In *2023 7th International Conference on Information Technology (InCIT)*, 2023, pp. 86-91.
11. S. Rallapalli, D. Hegde, and R. Thatikonda, "Feature selection based ensemble support vector machine for financial fraud detection in IoT," In *2023 International Conference on Evolutionary Algorithms and Soft Computing Techniques (EASCT)*, 2023, pp. 1-7. doi: 10.1109/easct59475.2023.10392566
12. X. Fu, and J. Su, "Feature selection based on recursive feature elimination for enterprise fraud detection," In *2024 10th International Conference on Systems and Informatics (ICSAI)*, 2024, pp. 1-6. doi: 10.1109/icsai65059.2024.10893814
13. I. D. Mienye, and Y. Sun, "A machine learning method with hybrid feature selection for improved credit card fraud detection," *Applied Sciences*, vol. 13, no. 12, p. 7254, 2023. doi: 10.3390/app13127254
14. M. Grossi, N. Ibrahim, V. Radescu, R. Loredo, K. Voigt, C. Von Altrock, and A. Rudnik, "Mixed quantum-classical method for fraud detection with quantum feature selection," *IEEE Transactions on Quantum Engineering*, vol. 3, p. 112, 2022.
15. S. Han, K. Zhu, M. Zhou, and X. Cai, "Competition-driven multimodal multiobjective optimization and its application to feature selection for credit card fraud detection," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, no. 12, pp. 7845-7857, 2022. doi: 10.1109/tsmc.2022.3171549