Proyecto de Machine Learning para la Detección de Fraude en Transacciones Financieras Móviles

Kevin Esteban Ruda Gómez*, Jeferson Alexander del Rio Herrera†, Juan Carlos Santa Hurtado‡
Departamento de Ingeniería de Sistemas, Universidad de Antioquia
Medellín, Colombia
Email: *kevin.ruda@udea.edu.co, †jeferson.delrio@udea.edu.co, ‡juan.santa1@udea.edu.co

***Resumen*—Este trabajo desarrolla un sistema de detección de fraude en transacciones de dinero móvil utilizando técnicas de Aprendizaje de Máquina. Se emplea el conjunto de datos "Fraud Detection Dataset", que contiene 6.3 millones de registros de transacciones. Se implementa un paradigma de aprendizaje supervisado con clasificación binaria para distinguir entre transacciones legítimas y fraudulentas, también se aborda el desafío del severo desbalance de clases (0.13 % de fraude) presente en los datos.**

***Palabras Clave*——Detección de fraude, aprendizaje de máquina, clasificación binaria, transacciones móviles, PaySim**

I. Introducción

Los sistemas de pago móvil han transformado las transacciones financieras, alcanzando $767 mil millones de dólares para 2020. Este crecimiento exponencial ha fomentado la inclusión financiera, pero también ha incrementado la exposición al fraude, representando una amenaza crítica para los servicios financieros digitales.

Los métodos tradicionales basados en reglas estáticas resultan insuficientes ante las tácticas evolutivas de los defraudadores. El Aprendizaje Automático (ML) se ha consolidado como herramienta fundamental por su capacidad para identificar patrones complejos y adaptarse dinámicamente a nuevos comportamientos fraudulentos.

Este proyecto desarrolla un sistema de detección de fraude utilizando el "Fraud Detection Dataset". Los principales desafíos incluyen el severo desbalance de clases (fraude <0.13 %), minimizar falsos positivos y negativos, e implementar un sistema computacionalmente eficiente para aplicación en tiempo real.

II. Descripción del problema

El fraude en dinero móvil aprovecha la velocidad y anonimato de plataformas digitales para ejecutar transferencias no autorizadas, retiros fraudulentos y esquemas de lavado de dinero. Una solución basada en ML permite analizar millones de transacciones en tiempo real, identificando patrones anómalos y adaptándose a nuevas tácticas mediante reentrenamiento continuo.

II-A. Composición de la base de datos

El "Fraud Detection Dataset" de Kaggle contiene datos del simulador PaySim, diseñado para replicar características estadísticas de transacciones reales de dinero móvil.

Número de muestras: 6,362,620 transacciones.

Número de variables: 11 variables que capturan información temporal, tipo, monto y saldos.

Descripción de las variables:

- **step:** Unidad de tiempo en la simulación, donde cada paso equivale a una hora. El dataset abarca 743 pasos temporales, equivalentes a aproximadamente 30 días de actividad transaccional.
- **type:** Tipo de transacción realizada. Variable categórica que toma cinco valores: CASH-IN (depósito de efectivo), CASH-OUT (retiro de efectivo), DEBIT (transferencia a cuenta bancaria), PAYMENT (pago por bienes/servicios), y TRANSFER (transferencia entre usuarios del servicio).
- **amount:** Monto de la transacción en unidades monetarias locales. Variable numérica continua con rango desde pequeñas cantidades hasta transferencias superiores a 90 millones.
- **nameOrig:** Identificador único alfanumérico de la cuenta que origina la transacción.
- **oldbalanceOrg:** Saldo inicial de la cuenta de origen antes de ejecutar la transacción.
- **newbalanceOrig:** Saldo final de la cuenta de origen después de ejecutar la transacción.
- **nameDest:** Identificador único alfanumérico de la cuenta destino de la transacción.
- **oldbalanceDest:** Saldo inicial de la cuenta de destino antes de recibir la transacción.
- **newbalanceDest:** Saldo final de la cuenta de destino después de recibir la transacción.
- **isFraud:** Variable objetivo binaria que indica si la transacción es fraudulenta (1) o legítima (0).
- **isFlaggedFraud:** Variable binaria que indica si la transacción fue marcada como sospechosa por el sistema de reglas básico implementado. Este sistema marca transferencias que superan 200,000 unidades monetarias en una sola operación.

Datos faltantes: El dataset no presenta valores faltantes o nulos en ninguna de sus variables, por lo que no se requiere implementar estrategias de imputación.

Análisis exploratorio: El dataset presenta un desbalance de clases extremadamente severo: transacciones legítimas 99.87 % (6,354,407), fraudulentas 0.13 % (8,213), proporción 1:774. El fraude se concentra exclusivamente en TRANSFER y CASH-OUT. La variable *amount* presenta distribución asi-

métrica con rango amplio, indicando necesidad de normalización.

II-B. Codificación de variables

Para el correcto procesamiento por los algoritmos de ML, se aplicarán las siguientes estrategias de codificación:

Variable	Tipo	Codificación
step	Numérico discreto	Sin codificación
type	Categórico nominal	One-hot encoding
amount	Numérico continuo	Normalización
nameOrig	Identificador	Label encoding
oldbalanceOrg	Numérico continuo	Normalización
newbalanceOrig	Numérico continuo	Normalización
nameDest	Identificador	Label encoding
oldbalanceDest	Numérico continuo	Normalización
newbalanceDest	Numérico continuo	Normalización
isFraud	Numérico binario	—
isFlaggedFraud	Numérico binario	—

Cuadro I
DESCRIPCIÓN DE VARIABLES Y ESTRATEGIA DE CODIFICACIÓN PROPUESTA

Justificación: One-hot encoding para *type* evita relaciones ordinales inexistentes. Normalización (StandardScaler/MinMaxScaler) esencial para variables de monto y saldo con rangos amplios, crítica para algoritmos sensibles a escala (redes neuronales, SVM, regresión logística). Label encoding para identificadores de alta cardinalidad (>6M valores). Se evaluará crear variables derivadas: inconsistencias en balances, agregaciones temporales, ratios monto/saldo.

II-C. Paradigma de aprendizaje

Se utilizará **aprendizaje supervisado** con **clasificación binaria** para predecir *isFraud*.

Justificación: (1) Disponibilidad de etiquetas confiables; (2) Problema inherentemente binario (fraudulenta/legítima); (3) Métricas estándar interpretables (precisión, recall, F1-score, AUC-ROC, balanced accuracy); (4) Enfoque dominante en literatura especializada. Se compararán múltiples algoritmos: regresión logística, k-NN, ensambles (Random Forest, XGBoost, Gradient Boosting), redes neuronales y SVM, buscando mejor balance entre detección (recall) y minimización de falsas alarmas (precisión).

Métricas de evaluación: Debido al severo desbalance de clases, se emplearán métricas robustas que no sean sensibles a la distribución: *Balanced Accuracy* (promedio de recall por clase, especialmente útil en datos desbalanceados), *Precision* (proporción de predicciones positivas correctas), *Recall* (capacidad de detectar fraudes), *F1-score* (media armónica de precision y recall), y *AUC-ROC* (área bajo la curva ROC, independiente del umbral de decisión). Balanced accuracy es particularmente relevante ya que pondera equitativamente ambas clases, evitando modelos sesgados hacia la clase mayoritaria.

III. ESTADO DEL ARTE

Se revisaron trabajos que abordan la detección de fraude financiero mediante técnicas de ML, enfocándose en problemas similares con datos desbalanceados.

Integración de grafos relacionales múltiples: Li y Yang [1] desarrollaron el modelo Tri-RGCN-XGBoost para detección de fraude financiero. *Paradigma:* Aprendizaje supervisado combinando Graph Neural Networks (RGCN) con XGBoost. *Técnicas:* Tres redes RGCN independientes (usuario-dispositivo, usuario-comerciante, usuario-dirección) fusionadas con XGBoost para clasificación. *Validación:* Datos reales de transacciones financieras. *Métricas:* Accuracy, precision, recall, F1-score, AUC. *Resultados:* Mejoras significativas vs. GBDT y GraphSage: +17.7 % en recall, +10.5 % en F1-score. Análisis SHAP mostró que la relación usuario-comerciante fue más influyente.

Ensambles para tarjetas de crédito: Khalid et al. [2] analizaron métodos supervisados para fraude en tarjetas de crédito europeas. *Paradigma:* Aprendizaje supervisado. *Técnicas:* K-NN, SVM, Decision Trees, Random Forest, Bagging, Boosting. Para el desbalance de clases aplicaron under-sampling y SMOTE. *Validación:* División 80 %-20 % (entrenamiento-prueba). *Métricas:* Accuracy, precision, recall, F1-score, ROC. *Resultados:* SMOTE mejoró significativamente la detección con accuracy entre 0.94-0.99 para todos los modelos evaluados.

REFERENCIAS

[1] J. Li and D. Yang, "Research on financial fraud detection models integrating multiple relational graphs," *Systems*, vol. 11, no. 11, p. 539, 2023. [Online]. Available: https://doi.org/10.3390/systems11110539

[2] A. R. Khalid, N. Owoh, O. Uthmani, M. Ashawa, J. Osamor, and J. Adejoh, "Enhancing credit card fraud detection: An ensemble machine learning approach," *Big Data and Cognitive Computing*, vol. 8, no. 1, p. 6, 2024. [Online]. Available: https://doi.org/10.3390/bdcc8010006