Mapping Illicit Flows: A Machine Learning Approach to Anti-Money Laundring

Emanuele Iaccarino[1] Giuseppina Iannotti[1] Alessia Migneco[1] Sara Pantini[1]

[1] Sapienza University of Rome

Abstract

Money laundering poses a significant threat to the integrity of global financial systems by enabling organized crime and terrorism. Detecting such illicit activities is challenging due to their rarity, delayed verification, and the deliberate mimicry of legitimate transactions. Traditional rule-based systems often fail to identify sophisticated laundering schemes, necessitating more advanced approaches. This work proposes an ensemble model combining XGBoost and LightGBM, leveraging their complementary strengths to improve detection performance. We focus on extracting structural and topological features from transactional data to uncover anomalous patterns indicative of money laundering. The proposed ensemble method is evaluated on both real and synthetic datasets, demonstrating strong accuracy, scalability, and generalization capabilities.

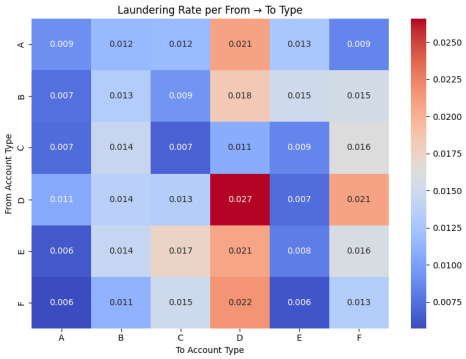
1 Introduction

Money laundering represents one of the most serious and persistent threats to the integrity of global economic and financial systems. Beyond undermining the stability of financial markets, it serves as a critical tool for organized crime and terrorist activities. Detecting money laundering activities presents numerous inherent challenges. These include the rarity of illicit events, the delayed confirmation of suspicious activities, and the deliberate effort by criminals to mimic legitimate financial behavior to evade detection. Effectively countering these practices requires the development of advanced tools capable of identifying atypical patterns and anomalous behaviors, often hidden within large volumes of financial data. Although many financial institutions currently rely on rule-based systems to generate Suspicious Activity Reports, such approaches often prove insufficient to detect laundering schemes. In response to these limitations, recent years have seen growing interest in machine learning techniques, which offer a more flexible and scalable framework. In this context, the present work explores innovative computational approaches for detecting suspicious money laundering activities, with particular focus on extracting structural and topological features capable of highlighting anomalies. These methods are evaluated in terms of accuracy, scalability, and generalization capabilities on both real and synthetic datasets. The goal of this work is to demonstrate the effectiveness of the proposed approach in identifying suspicious behaviors, thereby contributing to the fight against money laundering in the financial sector. The report is organized as follows. Section 3 presents the methodology. Section 4 shows the experimental results, including parameter initialization. Finally, Section 5 discusses the main results of the project and outlines possible future directions.

2 Key EDA Findings and DataEng

During the exploratory data analysis phase, we uncovered several meaningful behavioral patterns related to laundering activity. Each of these findings was translated into specific feature engineering choices that enriched our modeling strategy:

- **Directional Account Transitions:** We observed certain high-risk transactional flows.
- **High-Risk Payment Methods**
- **Self-Transfers Across Account Types:** Transfers made by users to themselves across different account types were not correlated at all with laundering.
- **Multi-Account User Profiles:** Users holding multiple account types under the same ID—up to



6 total—had a substantially higher probability of laundering activity.

- **Repetitive Peer Interactions:** We observed a clear pattern where users frequently sending or receiving funds from the same counterpart had a higher laundering probability. This lead to the inclusion of *in-degree* and *out-degree centrality* as node-level graph features.
- **Use of Intermediaries in Cycles:** Some users acted as intermediaries in laundering chains, forming cycles that eventually reached the final beneficiary. To detect this, we used betweenness centrality to identify structurally important nodes in laundering pathways.
- **Feature Transformation:** To ensure stability, we applied logarithmic transformations to the high-variance numerical features.

3 Method

This section details the machine learning pipeline developed for the detection of money laundering transactions. The proposed methodology combines structural insights derived from transaction graphs with predictive modeling using two gradient boosting algorithms: **XGBoost** and **LightGBM**. These models are independently trained and later fused through a weighted ensemble to strengthen predictive performance.

3.1 Graph-Based Feature Engineering

To capture the relational structure among entities in the transactional data, a directed graph $G = (V, E)$ was constructed, where each node $v \in V$ represents a financial account and each directed edge $e = (u \to v) \in E$ corresponds to a monetary transaction from account u to account v. From this graph structure, several features were derived. The *pair frequency*, defined as the number of transactions from u to v, is given by:

$$\text{pair\_frequency}(u, v) = |\{e \in E \mid e = (u \to v)\}| \quad (1)$$

To identify bidirectional patterns often associated with circular or reciprocal laundering, the *reverse pair frequency* was computed analogously:

$$\text{reverse\_pair\_frequency}(u, v) = |\{e \in E \mid e = (v \to u)\}| \quad (2)$$

Using these values, a binary feature was introduced to indicate whether a transaction was part of a circular structure:

$$\text{is\_circular}(u, v) = \begin{cases} 1 & \text{if reverse\_pair\_frequency}(u, v) > 0 \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

In addition, *betweenness centrality* was computed for each account to assess its role as an intermediary in the network. For a node v, its betweenness centrality is defined as:

$$C_B(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}} \quad (4)$$

where σ_{st} is the total number of shortest paths from node s to node t, and $\sigma_{st}(v)$ is the number of those paths that pass through v. This measure was mapped separately to the sender and receiver of each transaction, providing an additional layer of topological insight to the learning models.

3.2 Machine Learning Models

The classification problem was addressed using two supervised learning algorithms based on gradient boosting: **LightGBM** and **XGBoost**. These models were selected due to their proven effectiveness in handling high-dimensional structured data and imbalanced classification tasks. LightGBM, based on histogram binning and leaf-wise tree growth, enables fast training and high scalability. Its ability to construct deeper trees allows for learning fine-grained patterns in the data. Conversely, XGBoost grows trees in a level-wise fashion and includes L_1 and L_2 regularization terms to penalize model complexity. This helps mitigate overfitting and improves generalization. Both models were trained on the same feature set, which included graph-based and transactional attributes. Class imbalance, a common challenge in fraud detection, was explicitly addressed by setting the scale of positive class weights inversely proportional to their prevalence in the training data. The training process was conducted using stratified cross-validation to maintain consistent class distributions across folds.

3.3 Model Ensembling

To enhance robustness and leverage the complementary strengths of the two models, a probabilistic ensemble was constructed. The final predicted probability for each transaction was obtained by a convex

combination of the individual model outputs:

$$P_{\text{final}} = \alpha \cdot P_{\text{LGBM}} + (1 - \alpha) \cdot P_{\text{XGBoost}} \quad (5)$$

In this study, the ensemble weight was set to $\alpha = 0.4$, favoring the LightGBM model slightly less than XGBoost based on validation performance. This formulation benefits from LightGBM's ability to exploit local structures and XGBoost's strong regularization properties, leading to a more balanced and generalizable predictive model.

3.4 Final Prediction Pipeline

Following training, the ensemble model was applied to the test dataset, which had been preprocessed and feature-engineered in an identical manner to the training data. The ensemble output produced a probabilistic fraud score for each transaction. Final binary classification labels were obtained by applying a fixed decision threshold, and both the probabilities and class predictions were retained for downstream analysis and evaluation.

4 Experiments

To optimize our models for fraud detection under extreme class imbalance, we used **Optuna**, an automatic hyperparameter optimization framework. Specifically, we adopted the *Tree-structured Parzen Estimator* (TPE) sampler, a Bayesian optimization algorithm that models. At each iteration, TPE selects new hyperparameters by maximizing the expected improvement over the best score observed so far. Our objective function used 5-fold stratified cross-validation and returned a composite metric mentioned before.
We now present a comparative evaluation of the XGBoost and LightGBM models applied to our task. Both models were trained and tested under identical conditions. In particular, the models are compared across three key performance metrics: AUC, Balanced Accuracy, and Fraud Capture Rate. The results, summarized in Tab. 4, highlight the relative strengths of each model in detecting fraudulent activity under different evaluation perspectives. While both XGBoost and LightGBM performed strongly on their own, the ensemble model, proposed in 3, marginally improved the overall composite score. The combination benefitted from the precision of XGBoost and the deep tree learning of LightGBM. Notably, as shown in 4, the ensemble preserved the high fraud capture rate while slightly boosting AUC and balanced accuracy.

Table 1: Performance Comparison Between LGBM and XGBoost (Transposed)

Metric	XGBoost	LGBM
AUC	0.9987	0.9985
Balanced Accuracy	0.9972	0.9972
Fraud Capture Rate	0.9963	0.9963
Composite Score	0.9974	0.9973

Table 2: Performance Metrics Ensemble Model

Metric	Ensemble
AUC	0.90
Balanced Accuracy	0.79
Fraud Capture Rate	0.87
Composite Score	0.85

5 Conclusion

The preliminary evaluation on one-third of the test set yields a composite score of 0.71928, computed as the arithmetic mean of AUC, Balanced Accuracy, and Fraud Capture Rate. This result is particularly meaningful given the extreme class imbalance in the dataset, which renders standard evaluation metrics insufficient. The score reflects the model's strong discriminative power, its balanced performance across classes, and its effectiveness in prioritizing the most suspicious transactions for manual review. In summary, the results demonstrate that the model is not only capable of identifying laundering transactions effectively, but also aligns with the practical constraints and priorities of operational anti-money laundering processes.