Toronto Metropolitan University

Literature review and Exploratory Data Analysis

Detecting Money Laundering in Financial Transactions

Submitted by:
Avi Kaushik Student ID: 501338331

Supervisor:
Prof. Saman Hassanzadeh Amin

Date: June 24, 2025

Abstract

Despite significant progress in applying AI and machine learning to anti–money laundering (AML), financial institutions still struggle to keep pace with increasingly sophisticated laundering schemes. Traditional rule-based systems generate overwhelming numbers of false alarms and require constant manual rule updates, while supervised ML models, though more adaptive, often miss genuine illicit transactions due to extreme class imbalance and noisy, incomplete data. At the same time, advanced hybrid and graph-based approaches promise deeper insights into complex transaction networks but introduce new challenges around model interpretability and regulatory acceptance. Money laundering often hides in plain sight. Sophisticated laundering operations are designed to mimic normal transactional behavior, making detection extremely challenging. In this dataset, each row represents a transaction, accompanied by various attributes such as payment format, sending and receiving currency, amount, and a target label indicating whether the transaction was flagged as laundering. Moreover, real-time detection frameworks that integrate blockchain or smart-contract audit trails hold great promise for transparency and tamper resistance, yet they face practical hurdles such as high infrastructure costs, cross-border data-sharing restrictions, and strict privacy requirements. In this context, there is an urgent need for holistic AML solutions that can dynamically learn emerging laundering patterns, effectively address data imbalance without overfitting, and deliver clear, explainable risk scores that satisfy both compliance officers and regulators, all while preserving customer privacy and operational scalability.

1 Literature Review

1.1 Supervised Learning in AML: Tried and Tested, but Still Evolving

Many researchers continue to rely on supervised machine learning models, like decision trees, random forest, and gradient boosting, as they offer a solid balance between accuracy and interpretability. Lokanan (2022) for example, tested these models on a banking dataset and found that multi-layer perceptrons (a type of neural network) gave the best results for identifying money laundering behavior. Similarly, Harris et al. (2022) used real-world transaction data and found that traditional logistic regression, particularly when augmented with engineered features, outperformed more complex models like XGBoost in identifying money laundering risk.

Sintayehu and Seid (2023) proposed an AML identification model using five supervised machine learning algorithms, Random Forest, Decision Tree, SVM, Logistic Regression, and Naïve Bayes, on real banking data from the Commercial Bank of Ethiopia. After extensive preprocessing, including handling missing values and applying Synthetic Minority Oversampling Technique (SMOTE), they evaluated each model's performance. Random Forest outperformed the others with a striking 99.1% accuracy. Their study highlights the importance of data quality, feature selection via Pearson correlation, and the need for balanced datasets in achieving reliable AML detection. This approach illustrates how even in developing financial ecosystems, robust machine learning pipelines can significantly enhance fraud detection efforts.

Al-Ababneh et al. (2023) analyzed a dataset of 1 million transactions from Deutsche Bank using machine learning models for anti-money laundering. Among the tested models, Random Forest delivered the highest F1 score (0.88) and outperformed others in accuracy and sensitivity. The model was especially effective in identifying anomalies in international transfers, flagged as the highest-risk transaction type, highlighting its suitability for real-world financial monitoring applications.

Alkhalili et al. (2021) explored an often overlooked area in anti-money laundering, watch-list filtering, by building a machine learning system to help spot suspicious transactions more effectively. Their approach connected machine learning with existing systems through three simple steps: monitoring transactions, offering recommendations, and taking action when needed. Using real data from 1,500 blocked transactions, they tested different models like SVM, Naïve Bayes, and decision trees. The support vector machine, especially with a polynomial kernel, performed best, reaching 85% accuracy. What's more, the system helped cut down false alarms and reduced the workload for compliance teams. Their findings show how machine learning can make sanctions screening more efficient, especially when introduced gradually alongside traditional methods.

In another study, Correia (2024) addressed the common issue of unbalanced datasets (where fraudulent transactions are a tiny fraction of the total) by combining SMOTE with LightGBM. This allowed the model to correctly identify almost all laundering cases without being overwhelmed by noise.

Oztas et al. (2022) reviewed machine learning techniques used in AML transaction monitoring, categorizing them into five groups: historical case analysis, graph-based methods, anomaly detection, behavioral analysis, and risk ranking. Anomaly detection and behavioral modeling emerged as the most common, with algorithms like Isolation Forest and One-Class SVM showing strong performance. The authors highlighted issues such as high false positives in rules-based systems and scalability challenges in behavioral models, calling for better evaluation metrics, feature selection, and exploration of ensemble and deep learning approaches.

1.2 Beyond Basics: Hybrid Models and Graph-Based Detection

While supervised models are effective, they sometimes miss the bigger picture, especially when money laundering is hidden within complex webs of transactions. That's where hybrid and graph-based models come in.

Usman et al. (2024) proposed a system that uses a graph structure to model relationships between accounts and transactions. This setup lets the algorithm detect patterns like circular transfers or hidden links between entities, things that are hard to spot in traditional row-by-row data. Their graph-based model achieved high accuracy, even on a synthetic dataset designed to mimic real-world laundering networks.

Badal-Valero et al. (2023) took a creative route by combining Benford's Law (which looks for abnormal digit patterns in financial data) with machine learning. They used it as a first filter to flag suspicious data, then applied support vector machines and decision trees to zoom in on the real risks. The two-layer approach helped reduce false positives and saved computing time.

In the crypto world, Pettersson Ruiz and Angelis (2022) compared four machine learning models using a Bitcoin transaction dataset. Random forest had the best precision, but decision trees were easier to interpret, which matters when regulators need to understand why an account was flagged.

Ouf et al. (2023) proposed a full AML detection system that combines classification, clustering, and rule mining in a single pipeline. Their model, APPD-OML, was more accurate than using any one technique alone, showing that hybrid models often bring the best of all worlds.

1.3 Deep Learning: Great Power, Great Complexity

Several studies turned to deep learning, particularly neural networks, for their ability to capture complex patterns. Lokanan (2022) and Correia (2024) both found that neural networks, when trained properly, outperformed simpler models, especially in messy or nonlinear data.

Balani et al. (2024) used the popular PaySim dataset, which simulates mobile money transfers, to test various models including neural nets, logistic regression, and k-nearest neighbors. XGBoost stood out, with near-perfect accuracy, but simpler models like logistic regression were easier to train and explain.

Alotibi et al. (2022) investigated the application of deep learning and machine learning techniques to detect money laundering in cryptocurrency transactions, using the Bitcoin Elliptic dataset. They evaluated several algorithms including Deep Neural Networks (DNN), Random Forest (RF), K-Nearest Neighbors (KNN), and Naïve Bayes (NB). The Random Forest model achieved the highest performance with an F1-score of 0.99, closely followed by the Deep Neural Network, which obtained an F1-score of 0.98. The study demonstrated that both deep learning and traditional machine learning algorithms perform strongly on cryptocurrency transaction data, suggesting their potential utility for AML detection tasks.

In crypto analysis, Pettersson Ruiz and Angelis (2022) found that while advanced models had more predictive power, decision-makers and regulators still preferred models that were easier to understand. This trade-off between performance and transparency is something almost every paper touched on in one way or another.

1.4 Real-Time Detection and Blockchain Integration

Shafin and Reno (2025) developed a hybrid AML framework combining machine learning models with Hyperledger Fabric blockchain to automate compliance checks. Machine learning models, including autoencoders and decision trees, were used off-chain to detect suspicious transactions, and the results were logged on-chain for auditability. The system demonstrated strong classification performance, with AUROC scores up to 0.93, while highlighting the trade-off between blockchain transparency and throughput in a simulated transaction environment.

Canhoto (2021) adopted an affordances-based perspective to examine how machine learning technologies are leveraged in the fight against money laundering and terrorism

financing. Drawing on interviews with compliance professionals, the study identified key affordances such as data synthesis, anomaly detection, and process acceleration as critical enablers of ML systems in AML contexts. Notably, the study emphasized that the success of ML deployment depends not only on technical performance but also on organizational readiness and regulatory alignment. These findings highlight the importance of aligning technological capabilities with institutional processes to ensure effective and responsible adoption of AI in financial compliance.

While the potential is clear, the authors also pointed out that implementing blockchain-based AML systems isn't easy. Regulatory buy-in, infrastructure upgrades, and cross-border data sharing remain big hurdles. Still, it's an exciting direction for future AML innovation.

1.5 Common Challenges and What Comes Next

Across all 15 papers, a few recurring challenges stand out. First, class imbalance remains a major problem, fraudulent transactions are rare, so models can easily get overwhelmed by the "normal" data. Researchers like Correia (2024) used oversampling techniques to deal with this, but the risk of overfitting remains.

Explainability is another major concern. High-performing models like neural networks or gradient boosting are often black boxes, which makes them hard to use in regulated industries like finance. Authors like Harris et al. (2022) and Pettersson Ruiz and Angelis (2022) suggested using simpler models or explainability tools like SHAP to make model decisions more transparent.

Finally, there's the issue of data access. Only a few studies, like those by Harris et al. (2022) and Al-Ababneh et al. (2023), used real-world transaction data. Most others relied on synthetic or anonymized datasets due to confidentiality restrictions. Going forward, better partnerships between academia, regulators, and financial institutions could help researchers get access to more meaningful data while preserving privacy.

2 Exploratory Data Analysis

2.1 Data Overview and Preparation

I began by importing the dataset and performing a basic inspection. The dataset is large and well-structured, with no missing values and all major columns present. Most variables initially had the object data type, so I explicitly converted relevant ones to string to reflect their categorical nature more clearly. This also allowed for smoother visualizations and encoding later on.

	Timestamp	From Bank	Account	To Bank	Account.1	Amount Received	Receiving Currency	Amount Paid	Payment Currency	Payment Format	Is Laundering
0	2022/09/01 00:20	10	8000EBD30	10	8000EBD30	3697.34	US Dollar	3697.34	US Dollar	Reinvestment	0
1	2022/09/01 00:20	3208	8000F4580	1	8000F5340	0.01	US Dollar	0.01	US Dollar	Cheque	0
2	2022/09/01 00:00	3209	8000F4670	3209	8000F4670	14675.57	US Dollar	14675.57	US Dollar	Reinvestment	0
3	2022/09/01 00:02	12	8000F5030	12	8000F5030	2806.97	US Dollar	2806.97	US Dollar	Reinvestment	0
4	2022/09/01 00:06	10	8000F5200	10	8000F5200	36682.97	US Dollar	36682.97	US Dollar	Reinvestment	0
5	2022/09/01 00:03	1	8000F5AD0	1	8000F5AD0	6162.44	US Dollar	6162.44	US Dollar	Reinvestment	0
6	2022/09/01 00:08	1	8000EBAC0	1	8000EBAC0	14.26	US Dollar	14.26	US Dollar	Reinvestment	0
7	2022/09/01 00:16	1	8000EC1E0	1	8000EC1E0	11.86	US Dollar	11.86	US Dollar	Reinvestment	0
8	2022/09/01 00:26	12	8000EC280	2439	8017BF800	7.66	US Dollar	7.66	US Dollar	Credit Card	0
9	2022/09/01 00:21	1	8000EDEC0	211050	80AEF5310	383.71	US Dollar	383.71	US Dollar	Credit Card	0

Figure 1: Data Overview

```
data.info()
✓ 0.0s
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5078345 entries, 0 to 5078344
Data columns (total 11 columns):
 #   Column              Dtype
---  ------              -----
 0   Timestamp           object
 1   From Bank           int64
 2   Account             object
 3   To Bank             int64
 4   Account.1           object
 5   Amount Received     float64
 6   Receiving Currency  object
 7   Amount Paid         float64
 8   Payment Currency    object
 9   Payment Format      object
 10  Is Laundering       int64
dtypes: float64(2), int64(3), object(6)
memory usage: 426.2+ MB
```

Figure 2: Data Info

The Timestamp column was converted to a proper datetime format for future time-based analysis. After that, I reviewed the feature types and distributions to get a general feel for what kind of variables we're working with. Overall, the dataset includes categorical fields like Payment Format, Transaction Type, and Currency, as well as numeric fields like Amount Paid and Amount Received.

One initial insight was the wide range in transaction amounts, from very small to very large, which hinted at possible outliers or unusually structured transactions.

2.2 Target Variable Analysis

Before diving into feature analysis, I looked at the distribution of the target variable: Is Laundering. As expected, the data is highly imbalanced — a small percentage of transactions are flagged as laundering, while the vast majority are clean. This is common in fraud detection scenarios and it immediately signals the need for careful evaluation strategies later in modeling (e.g., resampling, special metrics).

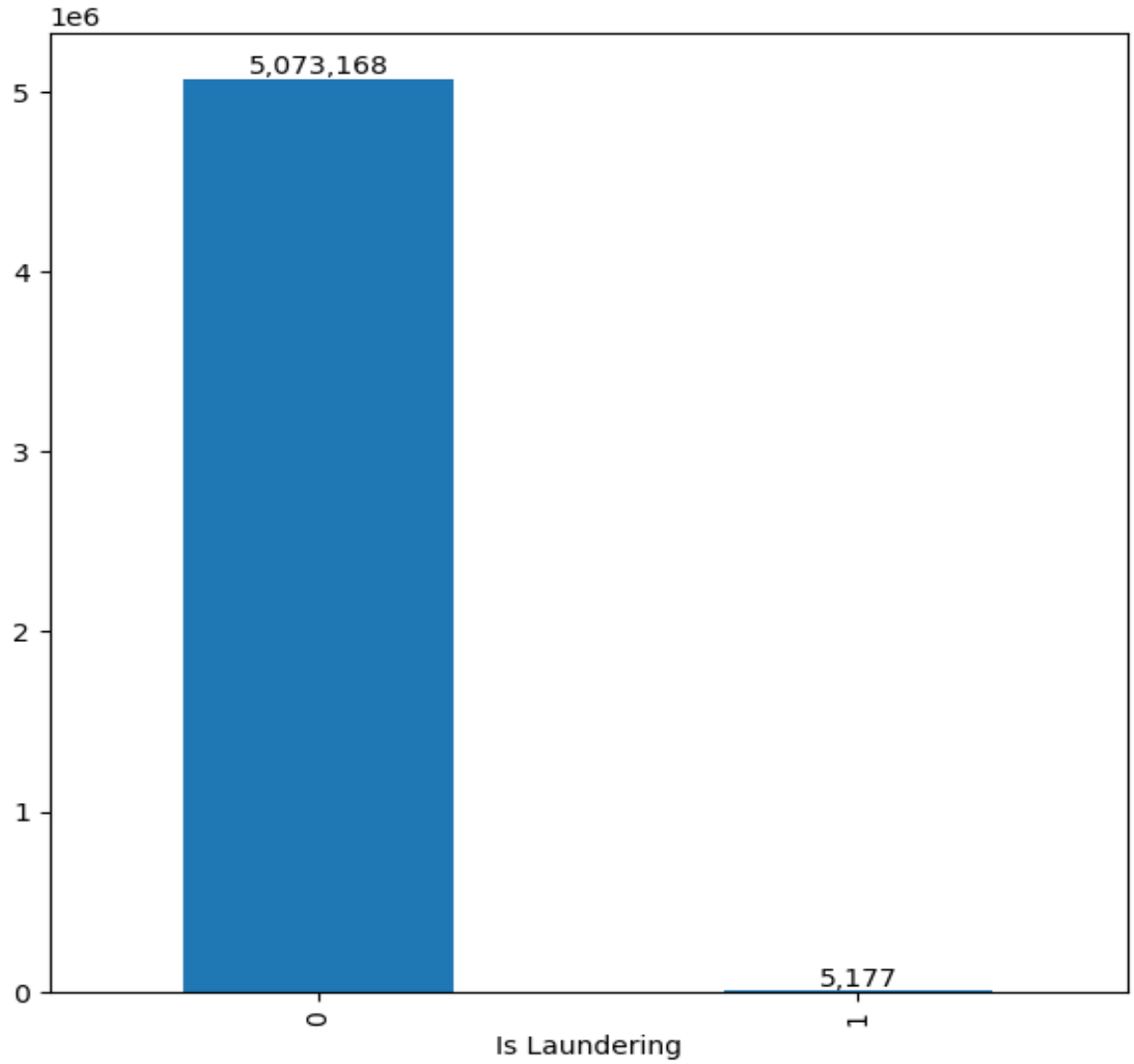


Figure 3: Is Laundering

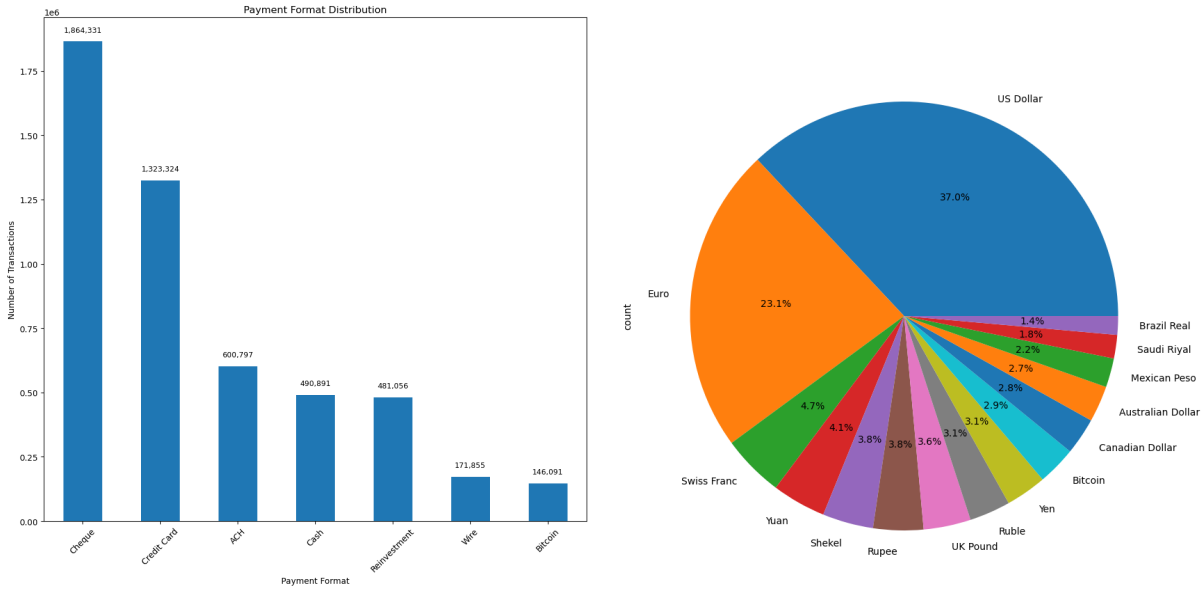
This imbalance also affects visualization. When plotting target counts, the laundering class barely registers unless scaled or explicitly highlighted. This reinforces the importance of using tools beyond simple class frequencies to identify meaningful differences in behavior.

2.3 Feature Distributions and Insights

2.3.1 Payment Format and Currency

I plotted distributions for some of the most informative categorical variables: Payment Format, Sending Currency, and Receiving Currency. Cheques and RTGS (Real-Time Gross Settlement) formats dominated the payment landscape, while IMPS and UPI had much smaller shares. This distribution aligns with real-world banking, where larger and business-related transactions prefer more traditional channels.

In terms of currency, INR (Indian Rupee), USD, and EUR were the most common. The dominance of these currencies isn't surprising given their global usage, but it could also mean laundering patterns might be disguised under the guise of international transactions.



(a) Payment-format (bar). (b) Payment-Currency (pie).

Figure 4: Distribution of payment formats.

2.3.2 Laundering Rate by Payment Format

I then calculated the proportion of laundering cases for each payment format. Instead of raw counts, I looked at the mean of the binary Is Laundering column grouped by format, which directly gives the laundering rate. This approach revealed that while

formats like Cheque and RTGS were commonly used, they also had slightly higher laundering rates than digital channels like UPI or NEFT.

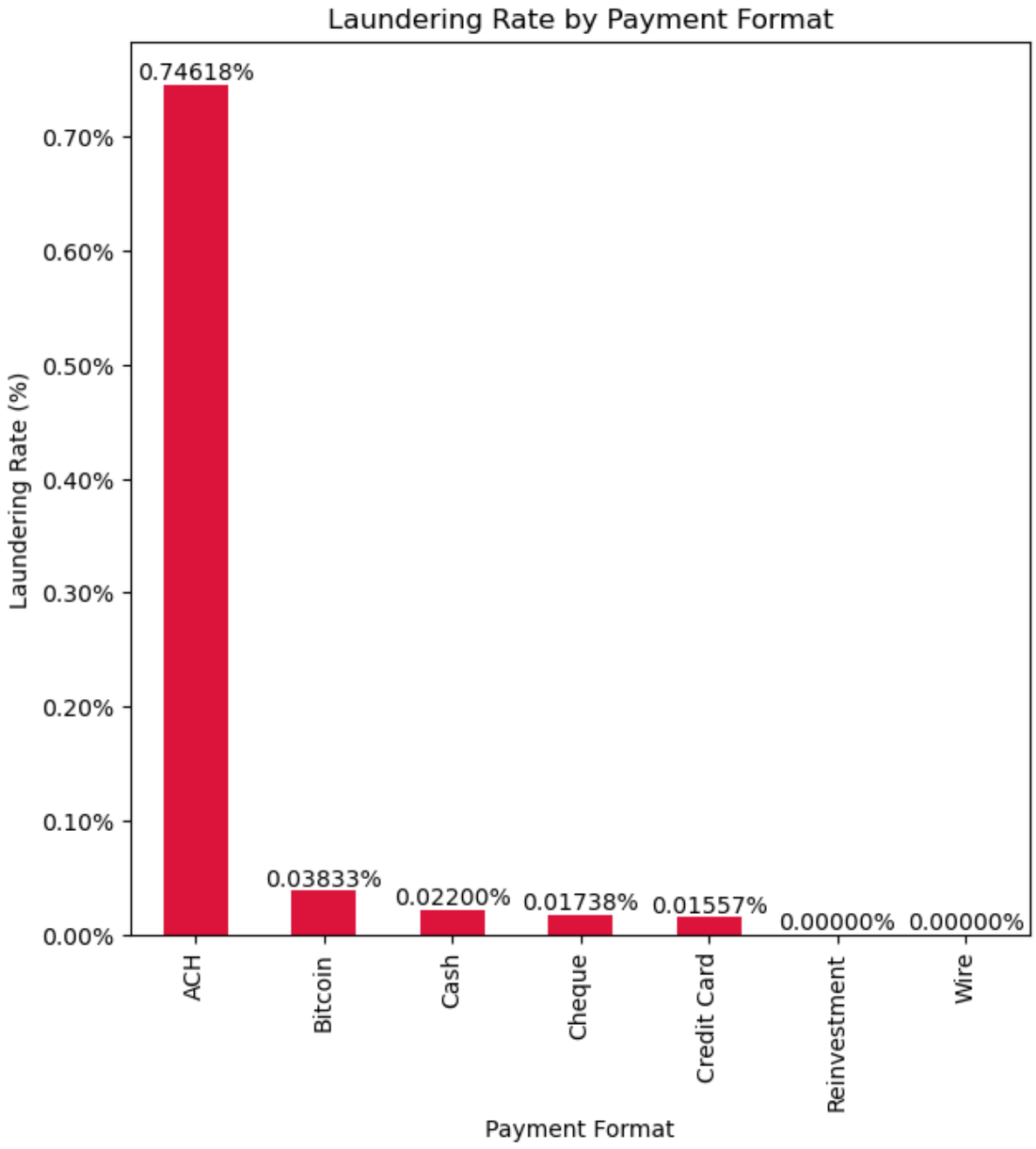


Figure 5: Laundering rate by payment format

This kind of insight is valuable, it suggests that even widely used methods aren't immune to misuse and may need closer inspection.

2.4 Numeric Feature Exploration

2.4.1 Correlation Heatmap

Next, I created a correlation heatmap using the numeric columns. The results showed very weak correlations between Amount Paid, Amount Received, and the laundering label. This suggests that the raw transaction amount, while often considered a red flag, isn't a strong standalone indicator in this dataset, likely because launderers keep amounts within "normal" ranges to avoid detection.

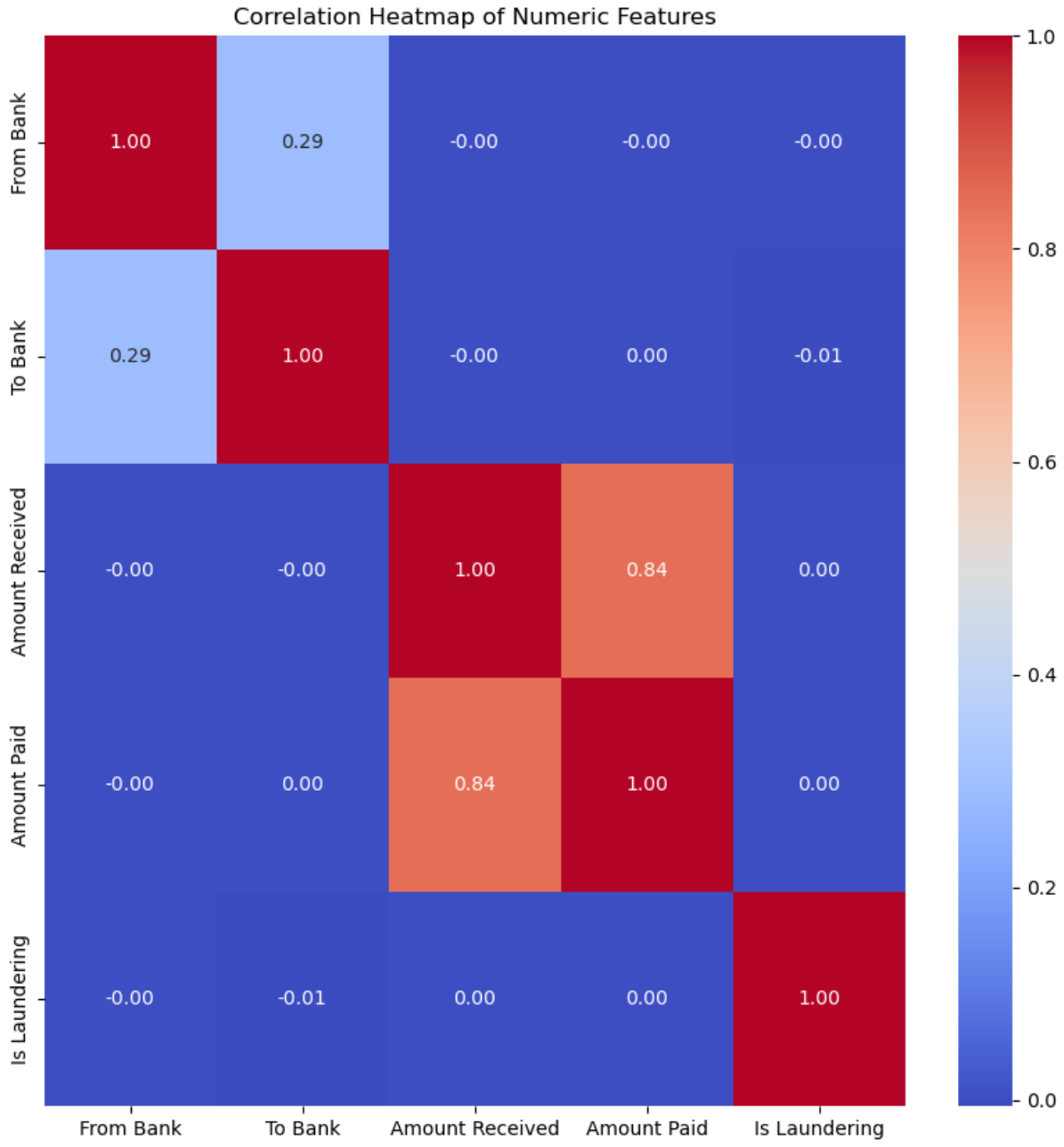


Figure 6: Correlation Heatmap

2.4.2 Mutual Information Scores

To overcome the limitations of correlation (which only captures linear relationships), I computed mutual information (MI) scores between each feature and the target. MI is more flexible and can uncover non-linear patterns.

This was a turning point: Payment Format and Receiving Currency showed the highest MI scores, confirming their importance in predicting laundering. Surprisingly, the amount-related features ranked lowest, further reinforcing the idea that launderers adapt by mimicking legitimate transaction sizes. This kind of insight guides us toward smarter modeling decisions, like deprioritizing or transforming raw amounts.

```
mi_df = pd.DataFrame({
    'Feature': X.columns,
    'MI Score': mi_scores
}).sort_values(by='MI Score', ascending=False)

print(mi_df)
```

✓ 0.0s Open 'mi_df' in Data Wrangler Python

Feature	MI Score
Payment Format	0.204051
Payment Currency	0.138437
Receiving Currency	0.138146
Account.1	0.001476
Account	0.001408
To Bank	0.000821
From Bank	0.000698
Amount Paid	0.000501
Amount Received	0.000472

Figure 7: MI Scores

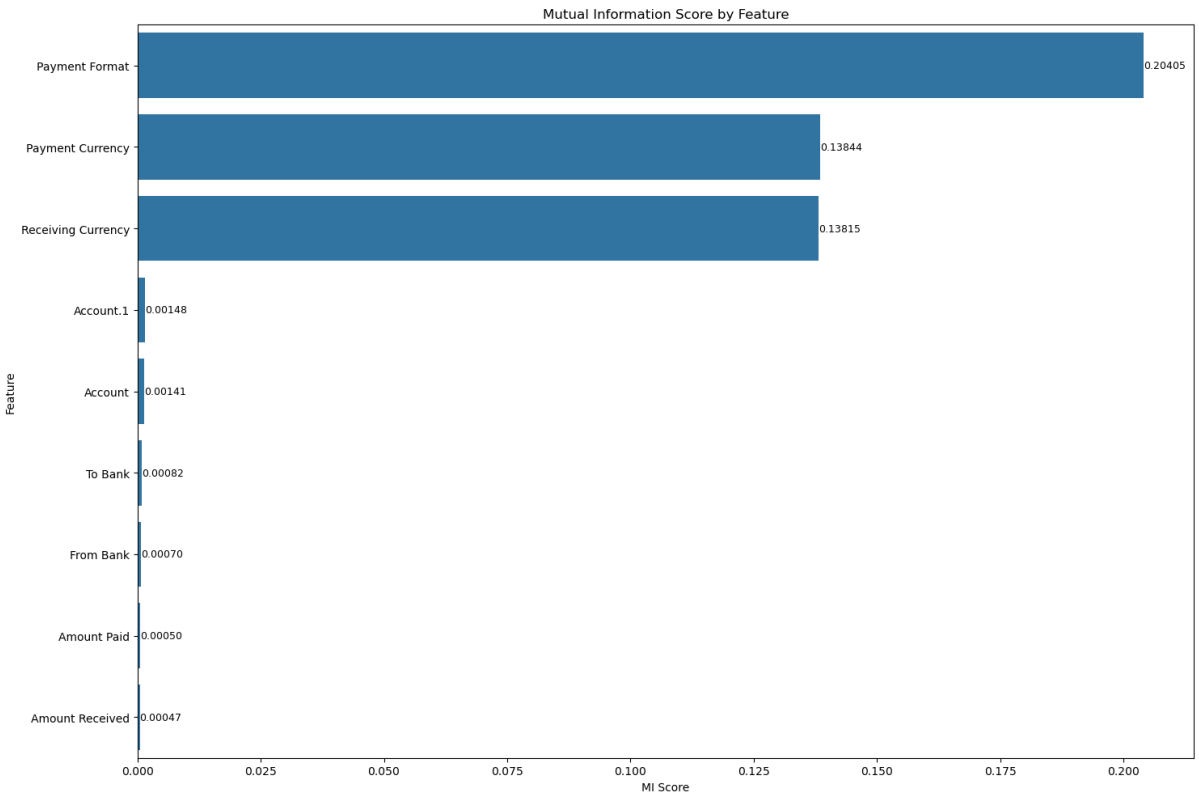


Figure 8: MI score by feature

2.5 Feature Engineering

Realizing that raw amounts weren't providing much value, I engineered several new features:

- **Log Transformation:** I applied log transformation (log1p) to both Amount Paid and Amount Received to reduce skewness and make the data more normally distributed. This helps with visualization and modeling.
- **Difference and Ratio:** I calculated the absolute difference and ratio between the two amounts. These could potentially highlight red flags, for example, large incoming funds followed by minimal outgoing ones (or vice versa) could suggest layering or smurfing behavior.

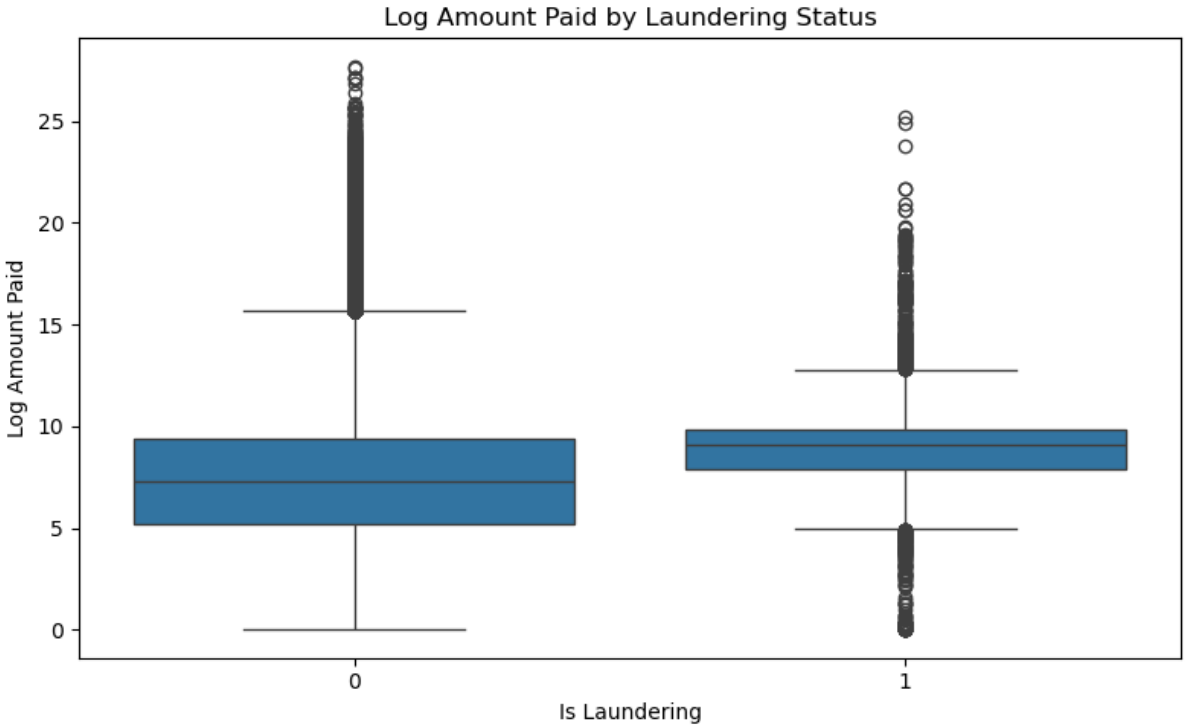


Figure 9: Log amount paid by laundering status

These engineered features offer better granularity and could serve as useful predictors in downstream models.

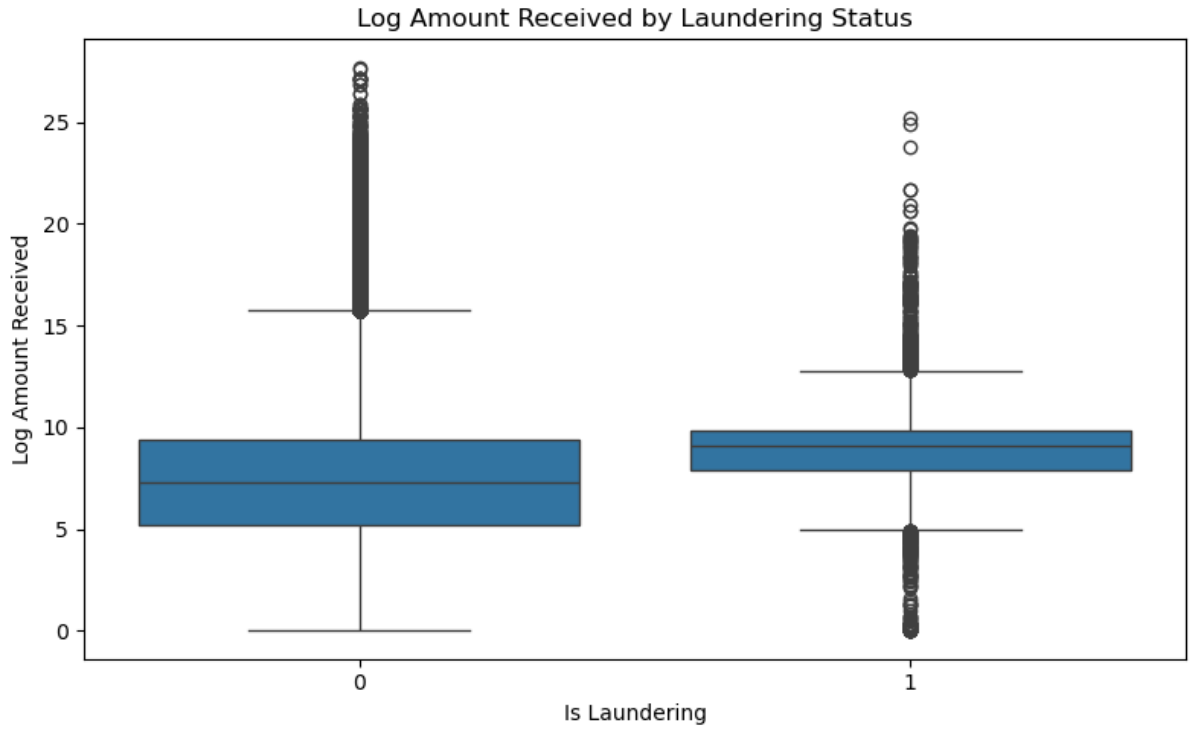


Figure 10: Log amount received by laundering status

2.6 Label Encoding for Modeling

To prepare the data for machine learning, I used LabelEncoder on categorical columns. This step is essential for algorithms like logistic regression or tree-based models that can't handle string inputs directly.

```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
categorical_columns = data.select_dtypes(include=['string']).columns
for col in categorical_columns:
    data[col] = le.fit_transform(data[col])
data.head()
```

✓ 12.5s 🐼 Open 'data' in Data Wrangler Python

	Timestamp	From Bank	Account	To Bank	Account.1	Amount Received	Receiving Currency	Amount Paid	Payment Currency	Payment Format	Is Laundering
0	2022-09-01 00:20:00	10	942	10	900	3697.34	12	3697.34	12	5	0
1	2022-09-01 00:20:00	3208	990	1	949	0.01	12	0.01	12	3	0
2	2022-09-01 00:00:00	3209	991	3209	940	14675.57	12	14675.57	12	5	0
3	2022-09-01 00:02:00	12	997	12	945	2806.97	12	2806.97	12	5	0
4	2022-09-01 00:06:00	10	999	10	947	36682.97	12	36682.97	12	5	0

Figure 11: Encoded Data

By encoding these variables numerically, we preserve their informational content while making them usable in both classical and modern modeling frameworks. I also ensured that timestamp-related features remained untouched for now, as I plan to extract components like day, month, and hour separately for time-based pattern detection.

3 Project Approach

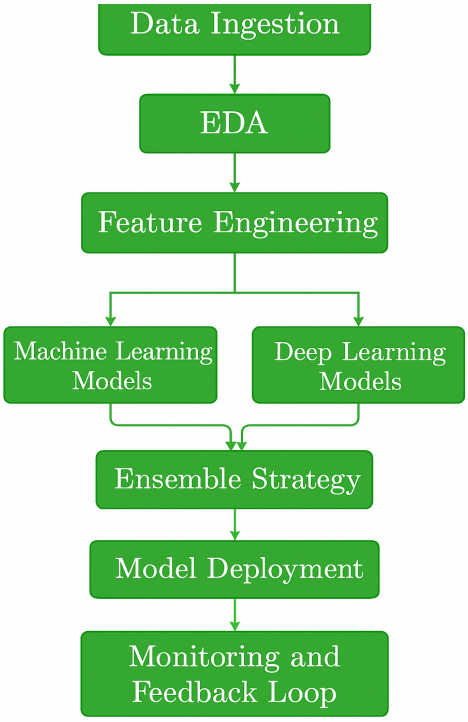


Figure 12: Project flow

This project followed a clear and structured process to analyze financial transaction data for signs of money laundering, as shown in Figure 12. It began with understanding and preparing the dataset, followed by exploratory data analysis (EDA) and feature engineering. The dataset was clean and well-organized, so no imputation or major corrections were needed.

During the EDA stage, various visualizations and statistical methods were used to uncover patterns in laundering activity, particularly in relation to currency types and payment formats. These insights helped guide feature engineering decisions. Transformations such as log scaling, amount ratios, and categorical encodings were applied to highlight useful signals in the data and improve model readiness.

Based on findings from the literature review, I will consider both machine learning and deep learning models. Algorithms such as Random Forests, Decision Trees, and Neural Networks will be selected for their strong performance in previous AML studies. Since no single model performs best in every case, I plan to use an ensemble strategy that combines multiple models to improve accuracy and stability.

To evaluate model performance, I will rely on metrics like precision, recall, F1-score, and AUC-ROC, which are particularly useful for detecting rare events such as money laundering. Once the models show promising results, I will prepare them for deployment within a financial monitoring system.

Following deployment, I intend to establish a monitoring and feedback loop. This will help ensure that the model continues to perform well over time and adapts to new patterns of laundering behavior. Additionally, I will use tools like SHAP to improve interpretability so that flagged transactions can be explained clearly and used with confidence by compliance teams.

4 GitHub Repository

I've uploaded everything for this project here on GitHub including the full literature review, all the EDA work, visualizations, and cleaned datasets. The project looks at how machine learning and deep learning can help detect money laundering in financial transactions. It's based on insights from 15 research papers and includes hands-on analysis of a real-world-style dataset. This repo will keep evolving as I continue building and testing models. Everything you see here is live and up-to-date.

`https://github.com/avikaushik282/Major-Research-Project`

References

N. Al-Ababneh, H. Al-Fetyani, N. Alabadi, and O. Al-Shamaileh. Money laundering detection using machine learning algorithms: A case study on deutsche bank. *International Journal of Engineering Research and Technology*, 12(3):478–483, 2023.

Sondos Alkhalili, Hassan Al-Bakri, and Samer AlHawari. Investigation of applying machine learning for watch-list filtering in anti-money laundering. *Future Internet*, 13(12): 324, 2021.

J. Alotibi, B. Almutanni, T. Alsubait, H. Alhakami, and A. Baz. Money laundering detection using machine learning and deep learning. *International Journal of Advanced Computer Science and Applications*, 13(10):732–738, 2022.

E. Badal-Valero, J. A. Alvarez-Jareño, and J. M. Pavía. Combining benford's law and machine learning algorithms for anti-money laundering. *Forensic Science International: Reports*, 12:101–107, 2023.

A. Balani, P. Gandhi, D. Agrawal, and A. Bhatia. Predicting fraud behaviour: A comparative analysis using machine learning on mobile money transactions. *Asian Journal of Computer Science and Technology*, 13(1):34–41, 2024.

Ana Isabel Canhoto. Leveraging machine learning in the global fight against money laundering and terrorism financing: An affordances perspective. *Journal of Business Research*, 131:428–437, 2021.

T. R. Correia. Predicting fraud behaviour – a data mining approach for anti-money laundering. *Insight Journal of Information Security*, 7(2):102–114, 2024.

J. Harris, S. Gad, N. Murtaza, and M. E. Lokanan. Using real-world transaction data to identify money laundering: Leveraging traditional regression and machine learning. *International Journal of Financial Studies*, 10(1):12, 2022.

M. E. Lokanan. Predicting money laundering using machine learning and artificial neural networks algorithms in banks. *Journal of Money Laundering Control*, 25(3):1–17, 2022.

A. Ouf, M. Ashraf, and M. Roshdy. A proposed paradigm using data mining to minimize online money laundering. *International Journal of Computer Applications*, 176(40): 11–19, 2023.

Berkan Oztas, Deniz Cetinkaya, Festus Adedoyin, and Marcin Budka. Enhancing transaction monitoring controls to detect money laundering using machine learning. *Journal of Money Laundering Control*, 25(2):362–375, 2022.

L. Pettersson Ruiz and L. Angelis. Combating money laundering with machine learning in the bitcoin ecosystem. *Digital Finance*, 4(3):185–201, 2022.

Khandakar Md Shafin and S. Reno. Integrating blockchain and machine learning for enhanced anti-money laundering system. *Journal of Artificial Intelligence and Blockchain Technology*, 8(2):89–103, 2025.

Kidist Sintayehu and Hussien Seid. Developing anti money laundering identification using machine learning techniques. *International Journal of Computer Applications*, 176(30): 5–10, 2023.

A. Usman, N. Naveed, and S. Munawar. Intelligent anti-money laundering fraud control using graph-based machine learning model for the financial domain. *International Journal of Security and Its Applications*, 18(2):22–36, 2024.