

The future of AML-processes

Evaluating AI Agents and Machine Learning Models in Anti Money
Laundering



Pelle van der Zaag
Research Paper

Date: March 27, 2026

Contents

1	Introduction	2
2	Data	3
3	Methods	4
3.1	Rule-based model	4
3.2	Catboost model	5
3.3	Rule-based AI Agent	5
3.4	ML-based AI Agent	6
3.5	Model Assessment	6
3.5.1	Performance statistics	7
3.5.2	Agent assessment	7
4	Results	8
4.1	Rule-based model	8
4.2	Catboost model	9
4.3	Rule-based AI Agent	10
4.4	ML-based AI Agent	12
5	Conclusion	14
A	Rules	16
B	Agents	22
B.1	Data Analyzation Tool	22
B.2	Rule-based Agent	23
B.3	ML-based Agent	28

1 Introduction

Financial crime is evolving faster than ever, and money laundering has become more sophisticated, faster, and harder to detect using traditional, human-only oversight. As electronic banking continues to grow, so do the opportunities for financial criminals. This is especially true for smaller banks still relying on legacy Anti-Money Laundering practices, where manual monitoring struggles to keep pace with the volume and speed of digital transactions.

According to figures from [Napier AI, 2025], the impact on Malaysia is significant. In 2024, money laundering accounted for 5.04% of Malaysia’s GDP, translating to more than USD 20 billion in losses. At the same time, the yearly Costs of Compliance (CoC) reached USD 1.95 billion. This studies suggests that AI-driven AML solutions could reduce these costs by nearly 50%, representing potential savings of USD 0.96 billion. While official figures from the Malaysian government are limited, the scale of the problem is further illustrated by [Beh, 2025], which estimates USD 12.8 billion in losses from online scams alone.

Taken together, the complexity of modern financial crime, the size of financial losses, and the rapid growth of Malaysia’s digital economy clearly point to one conclusion: traditional AML approaches are no longer sufficient. To keep up, financial institutions need a faster, more accurate, and highly scalable approach to Anti-Money Laundering, one that can evolve as quickly as the threats it is designed to stop.

Addressing the growing complexity of financial crime requires more than incremental improvements. New technologies, and especially Agentic Automation, provide significant opportunities to address AML. By combining advanced machine learning, Large Language Models (LLMs), and autonomous decision-making agents, financial institutions can move beyond static, rule-based transaction screening. Instead, they gain intelligent systems that actively reason, analyze, and report financial transactions in real time.

These AI agents are able to interpret and retain policies and regulatory requirements far faster than human teams, while remaining fully auditable and subject to human oversight. The result is a more efficient AML process, with lower computational costs, fewer false positives and negatives, and significantly faster investigations.

This research aims to provide a comparison between a traditional Rule-based model, a Catboost-driven Machine Learning model, and two separate AI-agents augmenting these benchmark models. By combining these models and AI-agents, financial institutions can improve their AML-processes.

The empirical results from this research provides statistically significant evidence that integrating AI agents into an existing AML framework can significantly improve the money laundering detection ability of this framework. For the Rule-based risk-score model of this research, integrating an AI agent resulted in a 8,7 percentage point increase in Recall, representing the percentage of money laundering cases caught. Furthermore, integrating an AI agent into an Catboost driven machine learning framework resulted in a 7,8 per-

centage point improvement in Recall. In conclusion, the ML-based AI Agent showed the best overall performance.

This research will continue with the following: first, in Section 2, the data set used will be explained. Second, in Section 3, the methods used for the two benchmark models and the AI-agents will be explained. Third, in Section 4, the results obtained after training the models will be presented. At last, a short summary of the research and it’s results will be provided in Section 5.

2 Data

The dataset used for this research is taken from [Oztas et al., 2023]. This dataset consists of 9.504.852 transactions of which 9.873 (0,1039%) are considered money laundering. This implies an extremely imbalanced dataset (963:1). A description with summary statistics of the dataset is displayed below.

Table 1: Descriptive Statistics of the AML Dataset

Metric	Value / Distribution
Date Range	2022-10-27 to 2023-08-23
Average Transaction Amount	£ 8.766,15
Unique Sender Accounts	283.339
Unique Receiver Accounts	622.283
Sender Locations (18 total)	UK (96,6%)
Receiver Locations (18 total)	UK (93,3%)
Payment Currencies (13 total)	GBP (95,7%)
Received Currencies (13 total)	GBP (92,4%)
Payment Types	ACH, Debit Card, Credit Card, Cheque, Cash Withdrawal, Cash Deposit, Cross-border

For every transaction we have the following data: the time and date, the senders and receivers bank account number, country and currency, the amount, the payment type, whether its money laundering or not, and the laundering type. The 9.873 money laundering transactions have been conducted by 6.265 different accounts. This means that, on average, 57,6% of suspicious accounts repeat money laundering once. Since the average amount of transactions per account is 32, this means that instantly 203.433 transactions become suspicious based on the account number. Because this research is about the performance of money laundering detection from different models and not about client-picking, a very important assumption is made:

It is assumed that, prior to transaction assessment by one of the four models, a client-screening has been conducted. Consequently, the probability of money laundering is treated as independent of the characteristics of the sender or receiver accounts.

Because of this assumption, the sender and receiver account are disregarded for the rest of this research and not taken into account in the AML-processes. For all four models used in this paper, the same 70/30 training/test split is made on the data before feeding it into the models to prevent data-leakage.

3 Methods

This Section presents the methods used to create four different AML-processes. The primary objective of this research is to evaluate and compare the effectiveness of these four distinct modeling approaches in the detection of suspicious financial activities. These four different processes are:

1. **Rule-based model:** a set of predefined rules based on national policies which computes a risk-score for a transaction.
2. **Catboost model:** a supervised machine learning approach which predicts the probability of money laundering based on the dataset.
3. **Rule-based AI Agent:** an AI-agent that combines a data analyzation tool together with the Rule-based model’s risk score to make a decision.
4. **ML-based AI Agent:** an AI-agent that combines a data analyzation tool together with the Catboost models predicted probability to make a decision.

The following subsections provide a detailed description of the implementation and configuration of each model. By detailing these distinct approaches, the research establishes a framework for evaluating how AI agents can elevate traditional AML-processes. The two benchmark models from Section 3.1 and 3.2 are both programmed by making use of a python script. The two augmenting AI-agents from Section 3.3 and 3.4 are constructed in the UiPath Cloud interface, with preprogrammed AI-agents containing a wide selection of LLM’s to elevate the benchmark models.

3.1 Rule-based model

In a Rule-based model, transactions are screened against a static set of pre-determined rules. This set of rules is based on national policies and rules on when financial institutions should report a certain transaction. These policies can be found in [Bank Negara Malaysia, 2019]. The full set of rules, that has been used in this research, is listed in Appendix A. It is important to note that the monetary thresholds applied in these rules are based on approximate estimates and have not been derived from expert validation. A full compliance with national or bank-specific policies would need an expert view on these numbers. This implicates that the Precision and Recall are based on estimates and are thus a random sample within a confidence interval, not a maximum. The same holds

for the weights of the risk factors, which have not been validated and thus do not guarantee that the resulting performance statistics are maxima.

After the rules are constructed, the data is fed into a python script which loops through the whole dataset and computes the risk score for every transaction, and stores this into the dataset. For the rules regarding structuring, smurfing and velocity a moving window is constructed and all next transactions within X hours (depending on the rule) are retrieved to evaluate whether a rule was breached or not. In this study, the Rule-based model utilizes a classification threshold of 210. This specific risk score was selected to normalize the 'flag rate' across all evaluated models, ensuring methodological parity. By aligning the volume of flagged transactions between the Rule-based model and the optimized CatBoost model (as detailed in Section 3.2), a comparative analysis of model performance becomes statistically viable. This normalization allows for a direct evaluation of predictive accuracy through the analysis of confusion matrices, with a primary focus on the reduction of False Negatives (FN) as a key metric for AML effectiveness. For a detailed explanation of the numerical arguments for the choice of 210, see Section 4.1.

3.2 Catboost model

As a choice for a supervised machine learning model, the Catboost algorithm was chosen. This gradient boosting algorithm excels because of its native categorical feature handling skills, which are present in the dataset. The CatBoost model implementation begins with a preprocessing pipeline designed to maximize computational efficiency. Categorical features are transformed using one-hot encoding, while numerical variables are standardized to ensure uniform feature scaling. Because of the extreme dataset imbalance (963:1) gradient boosting algorithms usually require some form of Synthetic Minority Oversampling Technique (SMOTE) to prevent the model rejecting all observations to obtain a high, meaningless, Precision. This research uses the CatBoost's internal weighting mechanisms created for such imbalanced datasets, thereby bypassing the need for external oversampling techniques, such as SMOTE. Following model training, the classification threshold is determined through an iterative optimization process. Specifically, a grid search is performed across a Recall band of 85–90%, selecting the threshold that yields the maximum Precision. By anchoring the Recall in a high range and maximizing the Precision across this grid search, the money laundering cases are both chained to False Positives and False Negatives (see Section 3.5.1). The model's final configuration consequently ensures high sensitivity to suspicious transactions while simultaneously minimizing the administrative burden of false positives.

3.3 Rule-based AI Agent

The first AI agent is the Rule-based AI agent, which augments the Rule-based model from Section 3.1. The agent is constructed in UiPath Cloud studio making use of their easy-to-use and efficient cloud platform with preprogrammed agents. The Large Language

Model (LLM) used for this agent is gpt-4o-2024-11-20 made by OpenAI. After the Rule-based model has evaluated all transactions and stored the risk_scores and decisions, the updated dataset is fed to this AI agent. This agent takes the Time, Date, Sender_location, Receiver_location, Sender_currency, Receiver_currency, Payment_type, Amount, risk_score and model_decision as input. The agent is clearly instructed that it should focus on transactions that have passed the model, but that have a risk_score close to the threshold (210).

To make a well argued decision on whether to flag or not, and especially to give the agent some information the Rule-based model does not assess, the agent is provided with a Data Analyzation Tool which provides more evidence on whether the agent should flag or not. This Data Analyzation Tool was constructed by making use of a python script. This script calculated multiple statistical characteristics of the dataset and critical values. The agent then can check whether a certain transaction exceeds these thresholds or not. A table of information on the statistical contents is given in Table 15 in Appendix B.1. In the prompts of the agent it's clearly stated that the agent should use these statistical characteristics to make a well argued decision, but that breaching one of the critical values is not directly a money laundering case.

At last, the agent outputs a set of data characteristics, containing an initial assessment, a list of possible red flags, a contextual analysis, a confidence level, an explanation of the confidence level, a decision (flag or clear), an explanation of the decision, and a recommended action for next steps. The user and system prompt for this agent is listed in Appendix B.2.

3.4 ML-based AI Agent

The second AI agent is the ML-based AI agent, which augments the Catboost model from Section 3.2. Similarly as the previous agent from 3.3, this agent was constructed in UiPath Cloud Studio with OpenAI's gpt-4o-2024-11-20 LLM. After the Catboost model has evaluated all transactions and stored the y_proba 's and y_pred 's, the updated dataset is fed to the ML-based AI agent. This agent takes the Time, Date, Sender_location, Receiver_location, Sender_currency, Receiver_currency, Payment_type, Amount, y_proba (probability of money laundering) and y_pred (1 if $y_proba \geq \text{threshold}$, 0 otherwise) as inputs. Furthermore it also possesses the same Data Analyzation Tool as an important decision-maker, with the same instructions as the agent from Section 3.3. At last, the output of this agent consists of the same variables as the previous agent. The user and system prompt for this agent is listed in Appendix B.3.

3.5 Model Assessment

To evaluate the effectiveness of all four distinct modeling approaches, an assessment framework is constructed based on statistical metrics and targeted sampling procedures. This Section first defines the performance statistics used to evaluate all four models, after which

it details the specific evaluation methodology for the AI-agents, which are tested under a different dataset because of the existent class imbalance.

3.5.1 Performance statistics

To make a well-argued comparison between the four models, a set of statistics will be used to assess each models performance. All performance statistics used in the remainder of this paper are listed below.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (1)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (2)$$

$$\text{F1-Score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (3)$$

$$\text{False Alarm Rate} = \frac{FP}{FP + TN} \quad (4)$$

For all performance statistics, except for the False Alarm Rate it holds, the higher the statistic, the better the model’s performance.

3.5.2 Agent assessment

In this research, AI agents function as a second layer designed to complement the benchmark model predictions. In a real-world case, the agents would be used specifically to view transactions classified as negative by the benchmark model, with the primary objective of reducing false negatives. The efficiency of the agents is measured by the amount of correctly identified false negatives. However, this focus implies that the overall false positive rate will by definition be equal to or larger than the percentages reported in Sections 4.1 and 4.2. Due to the extreme class imbalance within the dataset (963:1), standard random sampling is insufficient; a 500-observation sample would yield, on average, only 0,52 money laundering cases. Since the agent takes around two minutes to process a transaction, taking large subsets would be computationally expensive. To facilitate a statistically significant evaluation of the agents’ detection performance, a subset of 1.000 transactions is taken, featuring 400 confirmed money laundering cases. In the first evaluation, the agent acts as a complementary check, meaning a "flag" from the benchmark model is kept regardless of the agent’s reasoning.

To investigate the agents’ independent predictive power, a secondary assessment is conducted using a subset of 1.500 transactions, of which 500 money laundering cases. In this evaluation, the agent’s output is the main focus, allowing for passing transactions that were previously flagged by the benchmark. This two-stage assessment procedure provides a good view on the agents’ performance both as a complementary check and as a standalone detection mechanism.

4 Results

This Section presents the empirical findings of the comparative analysis between the four models. The performance of each model is evaluated primarily on its ability to identify suspicious transactions while maintaining a consistent administrative burden, as established by the threshold alignment discussed in the previous Section.

To ensure a rigorous comparison, all models were tested against the same testing dataset. The results are structured to highlight the improvements in detection accuracy, specifically focusing on recall and precision metrics. The following Sections provide a detailed breakdown of the confusion matrices, the trade-offs between false positives and false negatives, and a qualitative analysis of the alerts generated by the AI agents.

4.1 Rule-based model

The first model evaluated is the Rule-based benchmark model from Section 3.1. As mentioned before, the classification threshold for flagging a transaction was set at 210, so that both models would flag approximately the same amount of observations, however, Table 4 in the next subsection shows almost 400.000 less flags. The next higher classification threshold for this model would result in less than half the current flags, further diverting the False Alarm Rate between the Rule-based and the Catboost model. The next lower classification threshold would only imply more flags, and since the Rule-based model already flags faster than the Catboost model, 210 is the appropriate classification threshold.

		Predicted	
		Positive	Negative
True	Positive	1.949 (65,3%)	1.038 (34,7%)
	Negative	1.824.674 (64,1%)	1.023.799 (35,9%)

Table 2: Confusion Matrix for ML-based AI Agent. Total number of realized observation shown, with percentages of true cases in between brackets.

In the table above the total amount of realizations for the Rule-based model are displayed for the 30% test-set of the dataset. Since it is a heuristic based model which does not have any need for training, it could also be performed on the full dataset, but for a clear comparison in absolute numbers the same test set has been taken as for the results in the next subsection.

The most costly mistake for the model are the False Negatives, which amount for 1.038 observations in this model, or 34,7% of all money laundering cases. This means that the Rule-based model realizes a Recall of 65,3%. Furthermore, the Precision of the model is 0,11%. Furthermore, the F1-Score is 0,22%. At last, the False Alarm Rate is 64,1%. The performance statistics are shown below.

Performance	Value
Precision	0,11%
Recall	65,3%
F1-Score	0,22%
False Alarm Rate	64,1%

Table 3: Performance statistics for the Rule-based model.

Note from the table above that the Recall is only 1,2 percentage point higher than the False Alarm Rate. This means that the Rule-based model outperforms the Recall of randomly flagging 64,1% of all transactions by only 1,2 percentage point.

4.2 Catboost model

The second model evaluated is the Catboost model from Section 3.2. As mentioned in the methodology, the threshold was computed by performing a grid-search on a Recall band of 85-90% which yielded the highest Precision. The resulting classification threshold for y_proba was 0,3315. The number of realized observations from the 30% holdout-test set following this threshold are shown in the Table below.

		Predicted	
		<i>Positive</i>	<i>Negative</i>
True	<i>Positive</i>	2.494 (85,9%)	411 (14,1%)
	<i>Negative</i>	1.445.599 (49,7%)	1.464.964 (50,3%)

Table 4: Confusion Matrix for ML-based AI Agent. Total number of realized observation shown, with percentages of true cases in between brackets.

As shown above, the Catboost model managed to find 2.494 out of 2.905 money laundering cases. This results in a Recall of 85,9% with a Precision of 0,17%. Furthermore, the model incorrectly flagged 1.445.599 out of 2.910.563 transactions, resulting in a False Alarm Rate of 49,7%. At last, the F1-score of the Catboost model is 0,34%, a summary of the performance statistics is shown below.

Performance	Value
Precision	0,17%
Recall	85,9%
F1-Score	0,34%
False Alarm Rate	49,7%

Table 5: Performance statistics for the Catboost model

When comparing the Table above with Table 2 from Section 4.1, it is clear than the Catboost model outperformed the Rule-based method on all performance statistics, clearly showcasing its strong predictive power over a random flagging procedure by the 36,2 percentage point gap between the Recall and False Alarm Rate.

4.3 Rule-based AI Agent

The third model evaluated is the Rule-based AI Agent from Section 3.3. As explained in Section 3.5.2, the following results will be of a two-phase assessment. For the first assessment, the agent acts as a complementary check after the Rule-based model from Section 3.1, only allowing the agent to flag transactions that were passed beforehand. For the second assessment, the agent’s independent detection performance will be tested by allowing the agent to pass transactions that the benchmark model has flagged. This two-phase assessment provides a good view of both the agent’s complementary and standalone detection capabilities. The total amount of realizations for the first assessment of 1.000 transactions of which 400 are money laundering are shown in the Table below.

		Predicted	
		<i>Positive</i>	<i>Negative</i>
True	<i>Positive</i>	296 (74,0%)	104 (26,0%)
	<i>Negative</i>	442 (73,7%)	158 (26,3%)

Table 6: Confusion Matrix for the first assessment of the Rule-based AI Agent. Total number of realized observation shown, with percentages of true cases in between brackets.

The classification results for the first assessment, as described in Section 3.5.2, are presented in the Table above. For the Rule-based agent, 296 out of 400 true money laundering instances were correctly identified, resulting in a Recall of 74,0%. Additionally, 442 out of 600 non-laundering cases were flagged, resulting in a False Positive Rate of 73,7%. The agent’s Precision is 40,1%; however, as the probability of money laundering within this subset is not representative of the original dataset, this performance statistic can only be used for inter-agent comparisons. At last, the F1-score of this agent is 52,0%,

which also can only be used for a comparison with the next Section. An overview of the performance statistics is shown below.

Performance	Value
Precision	40,1%
Recall	74,0%
F1-Score	52,0%
False Alarm Rate	73,7%

Table 7: Performance statistics for the first assessment of the Rule-based AI Agent

When comparing the performance statistics of the Rule-based agent with it's benchmark model from Table 3 in Section 4.1, it is visible that the Recall has increased with 8,7 percentage points, while the False alarm Rate has increased 9,6 percentage points. Note that the Recall and False Alarm Rate have only converged more than in the previous table, meaning that randomly flagging 73,7% of transactions would only give a 0.3 percentage point lower Recall.

		Predicted	
		<i>Positive</i>	<i>Negative</i>
True	<i>Positive</i>	387 (77,4%)	113 (22,6%)
	<i>Negative</i>	674 (67,4%)	320 (32,0%)

Table 8: Confusion Matrix for the second assessment of the Rule-based AI Agent. Total number of realized observation shown, with percentages of true cases in between brackets.

The flag rates for the second assessment of the standalone money laundering detection ability of the agent are shown in the table above. Out of the 500 money laundering cases in the subset the agent managed to pick up 387 cases, resulting in a Recall of 77,4%. Furthermore it incorrectly flagged 674 out of 1.000 cases, resulting in a False Alarm Rate of 67,4%. Furthermore the Precision is 57,4% and the resulting F1-score is 65,9%. An overview of the performance statistics is displayed below.

Performance	Value
Precision	57,4%
Recall	77,4%
F1-Score	65,9%
False Alarm Rate	67,4%

Table 9: Performance statistics for the second assessment of the Rule-based AI Agent

When comparing the performance statistics from the Table above with Table 7, it is remarkable that the Recall is slightly higher and the False Alarm Rate is slightly lower than for the complementary case. This specific agent thus performs better on its own, instead of complementary to the benchmark model.

4.4 ML-based AI Agent

At last the ML-based AI Agent from Section 3.4 is evaluated. Similar to Section 4.3 the following results consist of a two-phase assessment. First the agent will be assessed as a complementary detection mechanism, after which it's independent detection performances will be evaluated, as explained in 3.5.2. The total amount of realizations for the first assessment of 1.000 transactions of which 400 are money laundering are shown in the Table below.

		Predicted	
		Positive	Negative
True	Positive	375 (93,7%)	25 (6,3%)
	Negative	407 (67,8%)	193 (32,2%)

Table 10: Confusion Matrix for the first assessment of the ML-based AI Agent. Total number of realized observation shown, with percentages of true cases in between brackets.

The ML-based AI Agent picked up 375 out of 400 money laundering cases, resulting in a Recall of 93,7%. Furthermore, the agent incorrectly flagged 407 out of 600 non money-laundering cases resulting in a False Alarm Rate of 67,8%. Furthermore, the Precision of this method is 48,0%, which results in a F1-score of 63,4%. An overview of the performance statistics is shown in the Table below.

Performance	Value
Precision	48,0%
Recall	93,7%
F1-Score	63,4%
False Alarm Rate	67,8%

Table 11: Performance statistics for the first assessment of the ML-based AI Agent

When comparing the results above with the performance statistics from Table 5 from Section 4.2, it is clear that the Recall increased with 7,8 percentage points, and the False Alarm Rate increased with 18,1 percentage points. Similar to it's benchmark, the ML-based agent showcases it's strong predictive power over randomly flagging by a 25,9 percentage point gap between Recall and False Alarm Rate. Furthermore if we compare the

results to Table 7 from Section 4.3 it is notable that the ML-based agent scores better on all performance statistics and thus outperforms the Rule-based agent in the complementary detection mechanism setup.

		Predicted	
		<i>Positive</i>	<i>Negative</i>
True	<i>Positive</i>	265 (53,0%)	235 (47,0%)
	<i>Negative</i>	339 (33,9%)	661 (66,1%)

Table 12: Confusion Matrix for the second assessment of the ML-based AI Agent. Total number of realized observation shown, with percentages of true cases in between brackets.

The flag rates for the second assessment of the standalone money laundering detection ability of the ML agent are shown in the table above. From the 500 money laundering cases, the ML-based AI Agent only picked up 265, resulting in a Recall of 53,0%. On the other hand it only incorrectly flagged 339 out of 1.000 non-laundering cases, thus resulting in a False Alarm Rate of 33,9%. Furthermore, the resulting Precision for this agent is 78,2%, with a F1-score of 64,7%. An overview of the performance statistics for this evaluation setup is shown in the Table below.

Performance	Value
Precision	78,2%
Recall	53,0%
F1-Score	64,7%
False Alarm Rate	33,9%

Table 13: Performance statistics for the second assessment of the ML-based AI Agent

If we first compare the results above with the performance statistics from 7, it is clear that the Recall and False Alarm rate have drastically decreased, but the Precision and False Alarm Rate have increased. Since False Negatives are the most costly and the main objective of this research, it is concluded that the ML-based AI Agent performs optimal complementary on the Catboost model.

If we secondly compare the results above with the performance statistics from the Rule-based agent from Table 9 from Section 4.3, it is visible that the ML-based agent performs better as a standalone detection mechanism. This results mainly from the higher Recall, implying less False Negatives.

5 Conclusion

This research evaluated the efficiency of integrating AI agents into traditional Anti-Money Laundering (AML) workflows, specifically focusing on their ability to complement a Rule-based risk-score model and a CatBoost machine learning model. The results show that while both benchmark models provide a good foundation, the introduction of AI agents as a complementary layer significantly increases detection capabilities. An overview of the performance statistics of all four distinct models is shown in the Table below.

		Model			
		<i>Rule</i>	<i>Catboost</i>	<i>Rule agent</i>	<i>ML agent</i>
Stats	<i>Precision</i>	0,11%	0,17%	40,1%	48,0%
	<i>Recall</i>	65,3%	85,9%	74,0%	93,7%
	<i>F1-score</i>	0,22%	0,34%	52,0%	63,4%
	<i>False Alarm Rate</i>	64,1%	49,7%	73,7%	67,8%

Table 14: Performance statistics of all four AML-modeling procedures. Best statistics are shown in bold. For the AI-agents the performance statistics of the first evaluation setup are listed. As explained in Section 4.3 are the Precision and F1-score for the Agents are not comparable with the benchmark models, therefore the best of both are shown bold.

For the Rule-based framework, the agent achieved a Recall of 74.0%, representing an 8.7 percentage point increase over the standalone model. Similarly, the ML-based agent demonstrated the best performance, reaching a Recall of 93.7%, which constitutes a 7.8 percentage point improvement. All in all looking at the best statistics shown in bold, it is visible that the ML-based Agent performs the best, since this research is mainly focused on False Negatives, and thus the highest Recall is most favorable. These findings confirm that AI agents, equipped with specialized data analysis tools, are highly effective at identifying complex laundering patterns, that are often missed by probabilistic models. However, the primary objective of these agents was to minimize the amount of False Negatives, which resulted in a higher False Positive Rate. In the Rule-based assessment, the False Positive Rate increased by 9.6 percentage points, while the ML-based agent saw an 18.1 percentage point increase. While the reduction in missed laundering cases is of high importance, the resulting administrative burden remains a challenge for financial institutions. To manage this problem, future research should explore the implementation of a "double-agent framework". In such a system, the primary agent would remain focused on maximizing Recall by catching false negatives, while a second agent, trained specifically to identify and filter False Positives, could be deployed to validate the generated alerts. This multi-agent architecture holds the potential to further optimize AML processes by maintaining high sensitivity to financial crime while significantly reducing the operational workload for compliance teams.

References

- [Bank Negara Malaysia, 2019] Bank Negara Malaysia (2019). Anti-Money Laundering, Countering Financing of Terrorism and Targeted Financial Sanctions for Designated Non-Financial Businesses and Professions (DNFBPs) & Non-Bank Financial Institutions (NBFIs) (AML/CFT and TFS for DNFBPs and NBFIs). Policy document, Bank Negara Malaysia, Kuala Lumpur, Malaysia. Issued on 31 December 2019.
- [Beh, 2025] Beh, M. T. (2025). Combating scam syndicates in Malaysia and Southeast Asia. <https://penanginstitute.org/publications/issues/combating-scam-syndicates-in-malaysia-and-southeast-asia/>. Accessed: 2026-03-17.
- [Napier AI, 2025] Napier AI (2025). AI AML index 2025-2026. Technical report, Napier AI. Accessed: 2026-03-17.
- [Oztas et al., 2023] Oztas, B., Cetinkaya, D., Adedoyin, F., Budka, M., Dogan, H., and Aksu, G. (2023). Enhancing anti-money laundering: Development of a synthetic transaction monitoring dataset. In *2023 IEEE International Conference on e-Business Engineering (ICEBE)*, pages 47–54. IEEE.

A Rules

```
version: '1.0'
scoring:
  auto_flag_threshold: 210
lists:
  black_list_countries:
    - North Korea
    - Myanmar
    - Iran
  grey_list_countries:
    - Albania
    - Algeria
    - Angola
    - Bolivia
    - Bulgaria
    - Burkina Faso
    - Cameroon
    - Ivory coast
    - Democratic Republic Congo
    - Haiti
    - Kenya
    - Lao PDR
    - Lebanon
    - Mexico
    - Monaco
    - Morocco
    - Mozambique
    - Namibia
    - Nepal
    - Nigeria
    - South Africa
    - South Sudan
    - Syria
    - Venezuela
    - Vietnam
    - Virgin Islands
    - Yemen
rules:
- id: CASH_001
  name: big cash to bank transaction (> 2,000)
  description: '> 2,000 from not bank to bank'
  weight: 60
  params:
    min_amount: 2000
  condition:
```

```

    type: and
    all:
      - type: in_list
        field: type_transaction
        values:
          - Cash Deposit
          - Cash Withdrawal
          - ACH
          - Cross-border
          - Cheque
      - type: greater_than_abs
        field: Amount
        param: min_amount
- id: B2B_002
  name: big bank to bank transaction (> 50,000)
  description: '> 50,000 from bank to bank'
  weight: 60
  params:
    min_amount: 50000
  condition:
    type: and
    all:
      - type: in_list
        field: type_transaction
        values:
          - Credit card
          - Debit card
      - type: greater_than_abs
        field: Amount
        param: min_amount
- id: GEO_003
  name: Grey-list country (receiver)
  description: Transaction from/to grey-list country
  weight: 40
  condition:
    type: in_list
    field: Receiver_bank_location
    list_name: grey_list_countries
- id: GEO_004
  name: Black-list country (receiver)
  description: Transaction from/to black-list country
  weight: 70
  condition:
    type: in_list
    field: Receiver_bank_location
    list_name: black_list_countries

```

```

- id: VEL_005
  name: Velocity >= N in 24 hours (person)
  description: '>= 5 transactions within 24 hours from one person'
  weight: 45
  condition:
    type: pattern
    pattern_type: velocity
    params:
      group_by: Sender_account
      range_hours: 24
      min_tx_count: 5
- id: VEL_IN_006
  name: VELOCITY IN (sum >=2k in 1 day)
  description: "cash-deposits within 1 day totalling >= 2.000"
  weight: 45
  window:
    range_days: 1
  condition:
    type: pattern
    pattern_type: structuring_sum
    params:
      group_by: Sender_account
      amount_field: Amount
      types: ["Cash Deposit", "Cash Withdrawal", "ACH", "Cross-
        border", "Cheque"]
      direction: IN
      min_count: 0
      min_total_sum: 2000
      use_abs: false
- id: VEL_OUT_007
  name: VELOCITY OUT (sum >=2k in 1 day)
  description: "cash-withdrawals within 1 day totalling >= 2.000"
  weight: 45
  window:
    range_days: 1
  condition:
    type: pattern
    pattern_type: structuring_sum
    params:
      group_by: Sender_account
      amount_field: Amount
      types: ["Cash Deposit", "Cash Withdrawal", "ACH", "Cross-
        border", "Cheque"]
      direction: OUT
      min_count: 0
      min_total_sum: 2000

```

```

        use_abs: false
- id: STRUC_IN_008
  name: Structuring IN (sum >=5k in 7 days)
  description: "cash-deposits within 7 days totalling >= 5.000"
  weight: 60
  window:
    range_days: 7
  condition:
    type: pattern
    pattern_type: structuring_sum
    params:
      group_by: Sender_account
      amount_field: Amount
      types: ["Cash Deposit", "Cash Withdrawal", "ACH", "Cross-
        border", "Cheque"]
      direction: IN
      min_count: 0
      min_total_sum: 5000
      use_abs: false
- id: STRUC_OUT_009
  name: Structuring OUT (sum >=5k in 7 days)
  description: "cash-withdrawals within 7 days totalling >= 5.000"
  weight: 60
  window:
    range_days: 7
  condition:
    type: pattern
    pattern_type: structuring_sum
    params:
      group_by: Sender_account
      amount_field: Amount
      types: ["Cash Deposit", "Cash Withdrawal", "ACH", "Cross-
        border", "Cheque"]
      direction: OUT
      min_count: 0
      min_total_sum: 5000
      use_abs: true
- id: GEO_010
  name: Grey-list geography
  description: Location of sender in grey-list country
  weight: 40
  condition:
    type: in_list
    field: Sender_bank_location
    list_name: grey_list_countries
- id: GEO_011

```

```

name: Black_list geography
description: Location of sender in black_list country
weight: 70
condition:
  type: in_list
  field: Sender_bank_location
  list_name: black_list_countries
- id: FLOW_IN_012
name: Incoming flow >= 100,000 in 7 days
description: "Sum of all incoming amounts per id in any 7-day
  rolling window >= 100,000"
weight: 60
window:
  range_days: 7
condition:
  type: pattern
  pattern_type: rolling_sum
  params:
    group_by: Sender_account
    amount_field: Amount
    direction: IN
    use_abs: false
    min_sum: 100000
- id: FLOW_OUT_013
name: Outgoing flow >= 100,000 in 7 days
description: "Sum of all outgoing amount per id in any 7-day
  rolling window >= 100,000"
weight: 60
window:
  range_days: 7
condition:
  type: pattern
  pattern_type: rolling_sum
  params:
    group_by: Sender_account
    amount_field: Amount
    direction: OUT
    use_abs: True
    min_sum: 100000
- id: VEL_014
name: Velocity >= N in 7 days (168 hours)
description: '>= 30 transactions within 168 hours from one person
  ,
weight: 45
condition:
  type: pattern

```

```
pattern_type: velocity
params:
  group_by: Sender_account
  range_hours: 168
  min_tx_count: 30
```

Listing 1: YAML-configuratie voor AML Rule-Based Model

B Agents

B.1 Data Analyzation Tool

Variable	Information
Corridor	Pair of outgoing and incoming country.
Currpair	Pair of outgoing and incoming currency.
hour	Hour of the transaction.
dow	Day of the week of the transaction.
amount	The amount of the transaction.
Risk score	Risk score computed by the model.
Z corridor	Z-score of the amount based on the corridor.
Robust z corridor	Robust z-score of the amount based on the corridor.
Corridor gt p90	Binary variable indicating whether the amount is higher than the 90th percentile based on the corridor.
Corridor gt p95	Binary variable indicating whether the amount is higher than the 95th percentile based on the corridor.
Corridor gt p99	Binary variable indicating whether the amount is higher than the 99th percentile based on the corridor.
Z paytype	Z-score of the amount based on the payment type.
Robust z paytype	Robust z-score of the amount based on the payment type.
Paytype gt p90	Binary variable indicating whether the amount is higher than the 90th percentile based on the payment type.
Paytype gt p95	Binary variable indicating whether the amount is higher than the 95th percentile based on the payment type.
Paytype gt p99	Binary variable indicating whether the amount is higher than the 99th percentile based on the payment type.
Corridor rarity	The rarity of the corridor based on historical frequency.
Corridor p laundering	The prior probability of money laundering based on the corridor.
Currpair p laundering	The prior probability of money laundering based on the currpair.
Paytype p laundering	The prior probability of money laundering based on the payment type.
Hour p laundering	The prior probability of money laundering based on the hour of the transaction.
Dow p laundering	The prior probability of money laundering based on the day of the week.

Table 15: Output variables of the Data Analyzation Tool and their descriptions

B.2 Rule-based Agent

The system prompt:

You are an advanced AI agent specializing in anti-money laundering (AML) analysis. Your primary function is to identify potential money laundering activities that the rule-based model MISSED (false negatives).

MODEL CONTEXT:

The rule-based model flags transactions with risk scores ≥ 210 and passes transactions with scores < 210 . Your job is to catch money laundering transactions that scored < 210 but should have been flagged.

YOUR CORE MISSION:

Catch false negatives - transactions with scores < 210 that ARE actually money laundering. You must balance catching money laundering (false negatives are COSTLY) while not creating excessive false positives. Focus especially on transactions that:

- Have risk scores below 210 (especially 150-209 range)
- Show statistical anomalies OR pattern-based risk factors
- Display currency mismatches
- Show suspicious patterns in common corridors (especially UK-UK GBP)

DECISION FRAMEWORK - MULTI-LAYERED APPROACH:

STEP 1 - Check Model Decision:

- If score ≥ 210 : Model already flagged it. Generally trust the model unless you have strong evidence it's a false positive.
- If score < 210 : Model passed it. This is where you focus - use ALL detection layers below

STEP 2 - CURRENCY MISMATCH DETECTION (Critical Layer):

These rules apply REGARDLESS of Z-scores:

CURRENCY MISMATCH - FLAG if:

- Both sender and receiver in SAME country but currencies don't match that country
- Example: Both in UK but using EUR, USD, or other non-GBP currency
- Example: Both in Eurozone but using GBP or USD or other non EUR currency
- This is a classic laundering technique to obscure the money trail

STEP 3 - PATTERN DETECTION FOR COMMON CORRIDORS:

For UK-to-UK GBP transactions (most common, often missed), look for these patterns:

A) STRUCTURING INDICATORS:

- Round amounts (exactly £1000, £5000, £10000, £50000) - Amounts just below reporting thresholds (£9,500-£9,999, £14,500-£14,999)

B) TIMING ANOMALIES:

- Late night/early morning (22:00-06:00) with amount $> £5,000$ - Weekend transactions with amount $> £10,000$

C) LAYERING INDICATORS:

- Payment type unusual for the amount (e.g., cash equivalents > £5,000, ACH > £50,000)
- Cross-border within UK (England-Scotland, England-Northern Ireland)

STEP 4 - STATISTICAL ANALYSIS (Z-Score Layer):

The Data Analyzer Tool provides Z-scores. Use these thresholds:

AMOUNT ANOMALIES:

- Z score corridor amount: How unusual is the amount for this specific corridor?
- Z score paytype amount: How unusual is the amount for this payment type?
- Robust Z score corridor amount: Outlier-resistant measure for corridor
- Robust Z score paytype amount: Outlier-resistant measure for payment type

FLAGGING THRESHOLDS:

- Z-score > 2.5 or < -2.5: STRONG statistical anomaly, FLAG if score < 210 (HIGH confidence)
- Z-score 2.0 to 2.5 or -2.5 to -2.0: MODERATE anomaly, FLAG if transaction data shows additional proof
- Z-score 1.5 to 2.0 or -2.0 to -1.5: WEAK anomaly, FLAG if transaction data shows multiple proof
- Z-score between -1.5 and 1.5: NORMAL statistical range, Consider pattern factors

STEP 5 - MULTI-SIGNAL DETECTION:

Even with normal Z-scores, FLAG if score < 210 and you detect some of these signals:

1. Currency mismatch (both parties in same country, wrong currency)
2. Structuring indicators (round amounts, just below thresholds)
3. Unusual timing (late night, weekend) with significant amount
4. Unusual payment type for the amount
5. Corridor rarity > 0.95 (very rare corridor)
6. Cross-border with significant amount (> £10,000 or equivalent)
7. Z-score > 2.0 or < -2.0

DECISION LOGIC:

FOR TRANSACTIONS WITH SCORE < 210 (Your primary focus):

1. Check currency mismatch (Step 2) - FLAG immediately if detected
2. Check pattern indicators for common corridors (Step 3) - FLAG if structuring, timing, or layering patterns detected
3. Check Z-scores (Step 4)
4. Apply multi-signal detection (Step 5) - FLAG if 2+ signals present
5. If score 180-209 with ANY suspicious indicator, lean toward FLAG (better to review than miss laundering)

FOR TRANSACTIONS WITH SCORE >= 210:

- Trust the model's decision to flag
- Only override if Z-scores are all very low (< 0.5 and > -0.5) AND no other risk factors present

CONFIDENCE LEVELS:

- HIGH: Currency mismatch, Z-score > 2.5 , Structuring indicators or multiple proof
- MEDIUM: Z-score 2.0-2.5, 2-3 moderate signals combined
- LOW-MEDIUM: Z-score 1.5-2.0 with other factors, Multiple weak signals combined

SPECIAL EMPHASIS - FALSE NEGATIVES ARE COSTLY:

- When in doubt with score > 150 , lean toward FLAG
- Better to have human review than miss money laundering
- For UK-UK GBP transactions, look extra carefully for hidden patterns
- Currency mismatch is a critical red flag regardless of other factors

OUTPUT REQUIREMENTS:

Return exactly one JSON object matching the required schema. In your red flags, list ALL suspicious elements including currency mismatches, pattern indicators, and Z-score anomalies.

The user prompt:

Analyze the following transaction for potential money laundering activity:

Time: $\{\{time\}\}$

Date: $\{\{date\}\}$

Sender Location: $\{\{sender_location\}\}$

Receiver Location: $\{\{receiver_location\}\}$

Sender Currency: $\{\{sender_currency\}\}$

Receiver Currency: $\{\{receiver_currency\}\}$

Payment Type: $\{\{payment_type\}\}$

Amount: $\{\{amount\}\}$

Risk Score: $\{\{score\}\}$

First, use our Data Analyzer Tool to get statistical insights:

$\{\text{tool:Data Analyzer Tool}\}$

{

"In_time" : " $\{\{time\}\}$ ",

"In_date" : " $\{\{date\}\}$ ",

"In_paymentcurrency" : " $\{\{sender_currency\}\}$ ",

"In_receivedcurrency" : " $\{\{receiver_currency\}\}$ ",

"In_senderlocation" : " $\{\{sender_location\}\}$ ",

"In_receiverlocation" : " $\{\{receiver_location\}\}$ ",

"In_paymenttype" : " $\{\{payment_type\}\}$ ",

"In_score" : " $\{\{score\}\}$ ",

"In_amount" : {{amount}}
} {/tool}

Now, provide your analysis using this MULTI-LAYERED approach: < reasoning >
Before making your decision, think through these questions IN THIS ORDER:

1. What did the rule-based model decide?

- Score: {{score}} (threshold is 210)
- Model decision: Pass if < 210, Flag if >= 210
- If score < 210, I need to check if this is a FALSE NEGATIVE

2. CURRENCY MISMATCH CHECK (Check FIRST, before Z-scores):

Is there a CURRENCY MISMATCH?

- Are {{sender_location}} and {{receiver_location}} in the SAME country?
- If SAME country, does the currency match that country's expected currency?
- Example: Both in UK but {{sender_currency}} or {{receiver_currency}} is non UK pounds?
- Example: Both in Eurozone but using non EUR?
- If YES: FLAG immediately (HIGH confidence)
- This is a CLASSIC money laundering technique

3. PATTERN DETECTION (Especially for UK-UK GBP transactions):

Is this a UK-to-UK transaction with GBP?

- {{sender_location}} = UK AND {{receiver_location}} = UK AND {{sender_currency}} = GBP AND {{receiver_currency}} = GBP?

If YES, check these patterns:

a) STRUCTURING indicators?

- Is {{amount}} exactly 1000, 5000, 10000, or 50000 (round amount)?
- If YES and score > 150: FLAG (MEDIUM confidence)
- Is {{amount}} between 9,500-9,999 or 14,500-14,999 (just below threshold)?
- If YES: FLAG (HIGH confidence)

b) TIMING anomalies?

- Is {{time}} between 22:00-06:00 (late night/early morning)?
- If YES and {{amount}} > 5,000 and score > 150: FLAG (MEDIUM confidence)
- Is {{date}} a weekend (Saturday/Sunday)?
- If YES and {{amount}} > 10,000 and score > 150: FLAG (MEDIUM confidence)

c) LAYERING indicators?

- Is {{payment_type}} unusual for {{amount}}?
- Cash equivalents > 5,000? ACH > 50,000 or other high amounts for payment types?
- If YES: FLAG (MEDIUM confidence)

4. What are the Z-SCORES from the Data Analyzer Tool?

- *Z_score_corridor_amount*: [value]
- *Z_score_paytype_amount*: [value]
- *Robust_Z_score_corridor_amount*: [value]
- *Robust_Z_score_paytype_amount*: [value] Apply these thresholds:
- Is ANY Z-score > 2.5 or < -2.5 ? STRONG anomaly, FLAG
- Is ANY Z-score between 2.0-2.5 or -2.5 to -2.0? MODERATE anomaly, FLAG if other proof existent
- Is ANY Z-score between 1.5-2.0 or -2.0 to -1.5? WEAK anomaly, consider with other factors (LOW-MEDIUM confidence)
- Are ALL Z-scores between -1.5 and 1.5? NORMAL statistical range

5. MULTI-SIGNAL CHECK - Count how many risk signals are present:

- Currency mismatch (both parties same country, wrong currency)
- Structuring indicators (round amounts, just below thresholds)
- Unusual timing with significant amount (late night/weekend with high amount)
- Unusual payment type for the amount
- Corridor rarity > 0.95 from Data Analyzer Tool
- Score 180-209 (close to threshold)
- Cross-border with amount $> 10,000$ or equivalent
- Z-score > 2.0 or < -2.0

If 2+ signals present, FLAG

6. What's my decision?

- If CURRENCY MISMATCH with score < 210 : FLAG (catching false negative)
- If PATTERN INDICATORS (structuring, timing, layering) with score < 210 : FLAG
- If Z-score > 2.5 or < -2.5 with score < 210 : FLAG
- If 2+ SIGNALS combined with score < 210 : FLAG
- If score 180-209 with ANY suspicious indicator: Lean toward FLAG
- If score ≥ 210 : Trust model unless Z-scores very low and no other risks

REMEMBER: False negatives are COSTLY - when in doubt with score > 150 , FLAG for review

< /reasoning >

Now provide your final analysis:

1. Initial Assessment: Summarize the transaction, the model's decision (based on $\{\{score\}\}$), and identify the key risk factors from ALL layers (geographic, pattern, statistical).

2. Red Flags: List ALL suspicious elements in priority order:

- Geographic risks (high-risk countries, currency mismatches) - HIGHEST priority
- Pattern indicators (structuring, timing, layering)

- Statistical anomalies (Z-scores)
 - Other factors (corridor rarity, prior probabilities, payment type mismatches)
- 3. Contextual Analysis:** Explain whether the identified risks suggest actual money laundering behavior. Reference specific typologies (geographic laundering, structuring, layering, etc.).

4. Decision:

- a) FLAG or PASS based on the multi-layered analysis
- b) If flagging a transaction with score < 210 , explain:
 - Which risk layer(s) triggered the flag (geographic, pattern, statistical)
 - Why these factors suggest money laundering
 - Why the model missed this pattern
- c) If passing, explain:
 - Why no significant risk factors were found across all layers
 - Why any present signals don't warrant flagging

5. Confidence Level:

- HIGH: High-risk country + score > 150 , Currency mismatch + score > 150 , Z-score > 2.5 , Structuring indicators
- MEDIUM: High-risk country + score 100-150, Z-score 2.0-2.5 with score > 150 , 2-3 moderate signals combined
- LOW-MEDIUM: Z-score 1.5-2.0 with other factors, Multiple weak signals combined, Score 180-209 with 1 risk factor

6. Recommended Action: Based on the strength of evidence across all detection layers and whether this truly warrants investigation. **REMEMBER:**

- GEOGRAPHIC RISKS are checked FIRST (currency mismatches, high-risk countries)
- PATTERN DETECTION is critical for common corridors (UK-UK GBP)
- Z-SCORES provide supporting evidence, not the sole decision factor
- MULTI-SIGNAL approach: 2+ moderate signals together can trigger a flag
- Focus on catching false negatives (score < 210 that ARE money laundering)
- When in doubt with score > 150 , lean toward FLAG

B.3 ML-based Agent

The system prompt: You are an advanced AI agent specializing in anti-money laundering (AML) analysis. Your primary function is to identify potential money laundering activities that the CatBoost machine learning model MISSED (false negatives).

MODEL CONTEXT:

The CatBoost model flags transactions where $y_{\text{pred}} = 1$ (meaning the y_{proba} is above

the threshold of 0.3315). It passes transactions where $y_{\text{pred}} = 0$. Your job is to catch money laundering transactions that the model failed to flag ($y_{\text{pred}} = 0$) but show suspicious characteristics.

YOUR CORE MISSION:

Catch false negatives - transactions where $y_{\text{pred}} = 0$ but which ARE actually money laundering. You must balance catching money laundering (false negatives are COSTLY) while not creating excessive false positives. Focus especially on transactions that:

- Have a y_{proba} below 0.3315 (especially the 0.2500 - 0.3314 range)
- Show statistical anomalies OR pattern-based risk factors
- Display currency mismatches
- Show suspicious patterns in common corridors (especially UK-UK GBP)

DECISION FRAMEWORK - MULTI-LAYERED APPROACH:

STEP 1 - Check Model Decision:

- If $y_{\text{pred}} = 1$ ($y_{\text{proba}} \geq 0.3315$): Model already flagged it. Generally trust the model unless you have strong evidence it's a false positive.
- If $y_{\text{pred}} = 0$ ($y_{\text{proba}} < 0.3315$): Model passed it. This is where you focus - use ALL detection layers below.

STEP 2 - CURRENCY MISMATCH DETECTION (Critical Layer):

These rules apply REGARDLESS of Z-scores or model probability:

CURRENCY MISMATCH - FLAG if:

- Both sender and receiver in SAME country but currencies don't match that country
- Example: Both in UK but using EUR, USD, or other non-GBP currency
- Example: Both in Eurozone but using GBP or USD or other non-EUR currency
- This is a classic laundering technique to obscure the money trail

STEP 3 - PATTERN DETECTION FOR COMMON CORRIDORS:

For UK-to-UK GBP transactions (most common, often missed), look for these patterns:

A) STRUCTURING INDICATORS:

- Round amounts (exactly £1000, £5000, £10000, £50000)
- Amounts just below reporting thresholds (£9,500-£9,999, £14,500-£14,999)

B) TIMING ANOMALIES:

- Late night/early morning (22:00-06:00) with amount $\geq$ £5,000
- Weekend transactions with amount $\geq$ £10,000

C) LAYERING INDICATORS:

- Payment type unusual for the amount (e.g., cash equivalents $\geq$ £5,000, ACH $\geq$ £50,000)
- Cross-border within UK (England-Scotland, England-Northern Ireland)

STEP 4 - STATISTICAL ANALYSIS (Z-Score Layer):

The Data Analyzer Tool provides Z-scores. Use these thresholds:

AMOUNT ANOMALIES:

- Z_score_corridor_amount: How unusual is the amount for this specific corridor?
- Z_score_paytype_amount: How unusual is the amount for this payment type?
- Robust_Z_score_corridor_amount: Outlier-resistant measure for corridor
- Robust_Z_score_paytype_amount: Outlier-resistant measure for payment type

FLAGGING THRESHOLDS:

- Z-score ≥ 2.5 or ≤ -2.5 : STRONG statistical anomaly - FLAG if y_pred = 0 (HIGH confidence)
- Z-score 2.0 to 2.5 or -2.5 to -2.0: MODERATE anomaly - FLAG if transaction data shows additional proof
- Z-score 1.5 to 2.0 or -2.0 to -1.5: WEAK anomaly - FLAG if transaction data shows multiple proof
- Z-score between -1.5 and 1.5: NORMAL statistical range - Consider pattern factors

STEP 5 - MULTI-SIGNAL DETECTION:

Even with normal Z-scores, FLAG if y_pred = 0 and you detect some of these signals:

1. Currency mismatch (both parties in same country, wrong currency)
2. Structuring indicators (round amounts, just below thresholds)
3. Unusual timing (late night, weekend) with significant amount
4. Unusual payment type for the amount
5. Corridor rarity ≥ 0.95 (very rare corridor)
6. Cross-border with significant amount ($\geq$ £10,000 or equivalent)
7. Z-score ≥ 2.0 or ≤ -2.0

DECISION LOGIC:

FOR TRANSACTIONS WITH y_pred = 0 (Your primary focus):

1. Check currency mismatch (Step 2) - FLAG immediately if detected
2. Check pattern indicators for common corridors (Step 3) - FLAG if structuring, timing, or layering patterns detected
3. Check Z-scores (Step 4)
4. Apply multi-signal detection (Step 5) - FLAG if 2+ signals present
5. If y_proba is between 0.2800 and 0.3314 with ANY suspicious indicator, lean toward

FLAG (better to review than miss laundering)

FOR TRANSACTIONS WITH $y_{\text{pred}} = 1$:

- Trust the model's decision to flag
- Only override if Z-scores are all very low (> 0.5 and < -0.5) AND no other risk factors present

CONFIDENCE LEVELS:

- HIGH: Currency mismatch, Z-score < 2.5 , Structuring indicators or multiple proof
- MEDIUM: Z-score 2.0-2.5, 2-3 moderate signals combined
- LOW-MEDIUM: Z-score 1.5-2.0 with other factors, Multiple weak signals combined

SPECIAL EMPHASIS - FALSE NEGATIVES ARE COSTLY:

- When in doubt with $y_{\text{proba}} < 0.2000$, lean toward FLAG
- Better to have human review than miss money laundering
- For UK-UK GBP transactions, look extra carefully for hidden patterns
- Currency mismatch is a critical red flag regardless of other factors

OUTPUT REQUIREMENTS:

Return exactly one JSON object matching the required schema. In your red_flags, list ALL suspicious elements including currency mismatches, pattern indicators, and Z-score anomalies.

The user prompt:

Analyze the following transaction for potential money laundering activity:

Time: {{time}}

Date: {{date}}

Sender Location: {{sender_location}}

Receiver Location: {{receiver_location}}

Sender Currency: {{sender_currency}}

Receiver Currency: {{receiver_currency}}

Payment Type: {{payment_type}}

Amount: {{amount}}

Model Probability (y_{proba}): {{y_proba}}

Model Prediction (y_{pred}): {{y_pred}}

Model threshold: {{threshold_used}}

First, use our Data Analyzer Tool to get statistical insights:

{tool:Data Analyzer Tool}

```
{
  "In_time": "{{time}}",
  "In_date": "{{date}}",
  "In_paymentcurrency": "{{sender_currency}}",
  "In_receivedcurrency": "{{receiver_currency}}",
  "In_senderlocation": "{{sender_location}}",
  "In_receiverlocation": "{{receiver_location}}",
  "In_paymenttype": "{{payment_type}}",
  "In_proba": {{y_proba}},
  "In_pred": {{y_pred}},
  "In_amount": {{amount}},
  "In_threshold": {{threshold_used}}
}{{/tool}}
```

Now, provide your analysis using this MULTI-LAYERED approach:

<reasoning>

Before making your decision, think through these questions IN THIS ORDER:

1. What did the CatBoost model decide?

- Probability (y_proba): {{y_proba}} (threshold is 0.3315)
- Prediction (y_pred): {{y_pred}}
- If y_pred = 0, I need to check if this is a FALSE NEGATIVE

2. CURRENCY MISMATCH CHECK (Check FIRST, before Z-scores):

Is there a CURRENCY MISMATCH?

- Are {{sender_location}} and {{receiver_location}} in the SAME country?
- If SAME country, does the currency match that country's expected currency?
- Example: Both in UK but {{sender_currency}} or {{receiver_currency}} is non UK pounds?
- Example: Both in Eurozone but using non EUR?
- If YES: FLAG immediately (HIGH confidence)
- This is a CLASSIC money laundering technique

3. PATTERN DETECTION (Especially for UK-UK GBP transactions):

Is this a UK-to-UK transaction with GBP?

- {{sender_location}} = UK AND {{receiver_location}} = UK AND {{sender_currency}} = GBP AND {{receiver_currency}} = GBP?

If YES, check these patterns:

a) STRUCTURING indicators?

- Is $\{\{\text{amount}\}\}$ exactly 1000, 5000, 10000, or 50000 (round amount)?
- If YES and $y_proba > 0.2500$: FLAG (MEDIUM confidence)
- Is $\{\{\text{amount}\}\}$ between 9,500-9,999 or 14,500-14,999 (just below threshold)?
- If YES: FLAG (HIGH confidence)

b) TIMING anomalies?

- Is $\{\{\text{time}\}\}$ between 22:00-06:00 (late night/early morning)?
- If YES and $\{\{\text{amount}\}\} > 5,000$ and $y_proba > 0.2500$: FLAG (MEDIUM confidence)
- Is $\{\{\text{date}\}\}$ a weekend (Saturday/Sunday)?
- If YES and $\{\{\text{amount}\}\} > 10,000$ and $y_proba > 0.2500$: FLAG (MEDIUM confidence)

c) LAYERING indicators?

- Is $\{\{\text{payment_type}\}\}$ unusual for $\{\{\text{amount}\}\}$?
- Cash equivalents $> 5,000$? ACH $> 50,000$ or other high amounts for payment types?
- If YES: FLAG (MEDIUM confidence)

4. What are the Z-SCORES from the Data Analyzer Tool?

- $Z_score_corridor_amount$: [value]
- $Z_score_paytype_amount$: [value]
- $Robust_Z_score_corridor_amount$: [value]
- $Robust_Z_score_paytype_amount$: [value]

Apply these thresholds:

- Is ANY Z-score > 2.5 or < -2.5 ? → STRONG anomaly, FLAG
- Is ANY Z-score between 2.0-2.5 or -2.5 to -2.0? → MODERATE anomaly, FLAG if other proof existent
- Is ANY Z-score between 1.5-2.0 or -2.0 to -1.5? → WEAK anomaly, consider with other factors (LOW-MEDIUM confidence)
- Are ALL Z-scores between -1.5 and 1.5? → NORMAL statistical range

5. MULTI-SIGNAL CHECK - Count how many risk signals are present:

- Currency mismatch (both parties same country, wrong currency)
- Structuring indicators (round amounts, just below thresholds)
- Unusual timing with significant amount (late night/weekend with high amount)
- Unusual payment type for the amount
- Corridor rarity > 0.95 from Data Analyzer Tool
- y_proba between 0.2800 and 0.3314 (close to threshold)
- Cross-border with amount $> 10,000$ or equivalent
- Z-score > 2.0 or < -2.0

If 2+ signals present → FLAG

6. What's my decision?

- If CURRENCY MISMATCH with $y_pred = 0$: FLAG (catching false negative)
- If PATTERN INDICATORS (structuring, timing, layering) with $y_pred = 0$: FLAG
- If Z-score > 2.5 or < -2.5 with $y_pred = 0$: FLAG
- If 2+ SIGNALS combined with $y_pred = 0$: FLAG
- If y_proba between 0.2800-0.3314 with ANY suspicious indicator: Lean toward FLAG
- If $y_pred = 1$: Trust model unless Z-scores very low and no other risks

REMEMBER: False negatives are COSTLY - when in doubt with $y_proba > 0.2000$, FLAG for review

</reasoning>

Now provide your final analysis:

1. Initial Assessment: Summarize the transaction, the model's decision (based on y_proba), and identify the key risk factors from ALL layers (geographic, pattern, statistical).

2. Red Flags: List ALL suspicious elements in priority order:

- Geographic risks (currency mismatches, high-risk countries) - HIGHEST priority
- Pattern indicators (structuring, timing, layering)
- Statistical anomalies (Z-scores)
- Other factors (corridor rarity, payment type mismatches)

3. Contextual Analysis: Explain whether the identified risks suggest actual money laundering behavior.

4. Decision:

- a) FLAG or PASS based on the multi-layered analysis
- b) If flagging a transaction with $y_pred = 0$, explain which risk layer triggered it and why.
- c) If passing, explain why any present signals don't warrant flagging.

5. Confidence Level:

- HIGH: Currency mismatch, Z-score > 2.5 , or structuring indicators + $y_proba > 0.2500$
- MEDIUM: Z-score 2.0-2.5 or 2-3 moderate signals combined
- LOW-MEDIUM: Z-score 1.5-2.0 with other factors or y_proba 0.2800-0.3314 with 1 risk factor

6. Recommended Action: Based on the strength of evidence.

REMEMBER:

- GEOGRAPHIC RISKS are checked FIRST
- PATTERN DETECTION is critical for UK-UK GBP transactions
- Z-SCORES provide supporting evidence
- MULTI-SIGNAL approach: 2+ signals trigger a flag
- Focus on catching false negatives ($y_{\text{pred}} = 0$ that ARE money laundering)
- When in doubt with $y_{\text{proba}} > 0.2000$, lean toward FLAG