International Journal of Management, Finance and Accounting

Identifying Potentially Illicit Money Laundering and Terrorism Financing Transactions Through Machine Learning Techniques

Shiuh Tong Lim[1,*], Rou Qing, Khoo[2], Khai Wah Khaw[1,], XinYing Chew[3]

[1]School of Management, Universiti Sains Malaysia, Gelugor, Malaysia
[2]Labuan Financial Services Authority, Financial Park Complex, Labuan, Malaysia
[3]School of Computer Sciences, Universiti Sains Malaysia, Gelugor, Malaysia
*Corresponding author: shiuhtong1997@student.usm.my (ORCiD:0009-0002-7741-9873)

Abstract

Financial institutions worldwide face significant challenges in detecting and preventing illicit financial activities, such as money laundering and terrorism financing. Traditional rule-based methods often generate high false positive rates, increasing manual verification efforts and higher operational costs. This research explores machine learning techniques to enhance the detection of suspicious transactions. Several algorithms, including K-Nearest Neighbors, Decision Tree, Random Forest, Logistic Regression, Support Vector Machine, and Naïve Bayes, are applied and evaluated using a dataset from a financial institution. After a comprehensive performance assessment, the Random Forest model is the most effective, exhibiting the highest accuracy of 0.9333 in identifying suspicious transactions while minimising false positives. These findings highlight the potential of integrating machine learning into financial crime prevention protocols. It serves as a guide for practitioners to predict suspicious transactions in financial institutions based on previous patterns of transactions. It also helps financial institutions to reduce compliance costs, which are typically higher than those of standard rule-based systems. However, this work presents a suspicious transaction prediction paradigm from prior behaviour with no transparency in features and with high accuracy and zero false positives, as with the Financial Action Task Force (FATF) promotion of new Anti-Money Laundering and Countering the Financing of Terrorism (AML/CFT) initiatives.

International Journal of Management, Finance and Accounting
Vol. 6, Issue 2 (2025) https://doi.org/10.33093/ijomfa.2025.6.2.9

Published by MMU PRESS. URL https://journals.mmupress.com/ijomfa

Keywords: Financial Institution, Machine Learning, Money Laundering, Suspicious Transaction Prediction, Terrorism Financing Transaction

Received on 7 March 2025; Accepted on 23 May 2025; Published on 30 August 2025

To cite this article: Lim, S. T., Khoo, R. Q., Khaw, K. W., & Chew, X. (2025). Identifying potentially illicit money laundering and terrorism financing transactions through machine learning techniques. *International Journal of Management, Finance and Accounting*, *6*(2), 243-267. https://doi.org/10.33093/ijomfa.2025.6.2.9

1.0 Introduction

Money laundering and terrorism financing (ML/TF) are pervasive financial crimes with significant socioeconomic consequences, exploiting vulnerabilities in an increasingly interconnected world. As criminal networks continuously adapt, financial institutions, regulators, and law enforcement agencies face growing challenges exacerbated by globalisation and technological advancements (IMF Staff, 2023). In response, the Financial Action Task Force (FATF) established international standards, complemented by Malaysia's enactment of the Anti-Money Laundering, Anti-Terrorism Financing, and Proceeds of Unlawful Activities Act 2001 (AMLA) (Bank Negara Malaysia, 2001; Financial Action Task Force, 2022) and Bank Negara Malaysia's (BNM) issuance of regulatory guidelines (Bank Negara Malaysia, 2019). With Malaysia recording a staggering 3.9 billion online transactions in 2020 (Statista, 2021)—primarily through e-money, internet banking, and mobile banking—compliance with BNM's Anti-Money Laundering and Countering the Financing of Terrorism (AML/CFT) transaction monitoring requirements presents a formidable challenge for financial institutions in combating illicit financial flows.

Financial institutions employ various methods to comply with transaction monitoring requirements, ranging from manual scrutiny to rule-based systems (Alexandre & Balsa, 2016; Gao & Ye, 2007; Omoseebi et al., 2025). While rule-based systems allow institutions to establish pre-defined thresholds for flagging suspicious transactions, they often generate high false-positive rates (Grint et al., 2017; Omoseebi et al., 2025). Monitoring these alerts is costly and time-consuming, diverting resources from verified cases that must be thoroughly analysed. In addition, using a one-size-fits-all approach to risk situations diminishes visibility into individual customer transactions, hindering effective monitoring and mitigation of ML/TF risks. As a result, financial institutions must balance cost and efficiency and turn to new compliance strategies. With growing amounts of data, companies and institutions seek ways to extract useful business insights within complex data layers. Machine learning-based data analytics gives powerful means for discovering value patterns that help businesses find new opportunities, read trends, and solve issues (Gandhi et al., 2024).

Adding machine learning to AML/CFT transaction monitoring can significantly enhance detection capability by leveraging history to identify complex transaction patterns that might evade manual detection. Machine learning-based algorithms are superior to rule-based systems since they learn to adapt to evolving ML/TF strategies, constantly updating with new data to optimise accuracy while minimising false positives and false negatives. By analysing different data sources, including customer profiles, machine learning enhances risk assessment and enables analysts to prioritise high-risk cases more effectively. By applying machine learning models in AML/CFT compliance, processes are automated, and resource utilisation is also optimised, reducing operational weights and total costs for financial institutions. In short, this study aims to predict suspicious money laundering and terrorist financing (ML/TF) transactions from large datasets using supervised machine learning methods. The goal is to enhance the accuracy of identifying suspicious activities and reduce false positives compared to traditional rule-based systems. Specifically, the objectives are to (1) develop a machine learning model for predicting suspicious ML/TF transactions, (2) identify and propose the most suitable model, and (3) provide financial institutions with an effective transaction monitoring solution to help combat illicit financial flows. The following sections present the literature review regarding suspicious ML/TF transaction detection. The research methodology is described in the third section. Then, the research findings and discussion are presented in the fourth section. Finally, the conclusion with implications, limitations, and future work is provided.

2.0 Literature Review

This section explains the literature review conducted for this study, which demonstrates machine learning algorithms used in previous research and how effective they proved to be in detecting suspicious Money laundering and terrorism financing (ML/TF) transactions. The second section also provides a comparative review of some algorithms.

2.1 Suspicious Money Laundering and Terrorism Financing (ML/TF) Transaction Detection

A literature review reveals a dearth of studies on ML/TF detection techniques, highlighting the urgent need for efficient, science-driven AML/CFT controls (Jullum et al., 2020; Ngai et al., 2011). This section briefly describes existing methodologies and their effectiveness in combating this widespread financial issue, as summarised in the following sub-sections.

2.1.1 Data Management

A broad survey examined various machine learning models, including support vector machines and decision trees for detecting suspicious transactions (Chen et al., 2018). One of the challenges encountered was the highly imbalanced nature of ML/TF datasets. The study indicated that such a challenge can be overcome by employing correct data refinement techniques, such as SMOTE or several resampling approaches, significantly enhancing the learning process and detection capability (Estabrooks et al., 2004; Ramentol et al., 2011).

2.1.2 Application of Unsupervised Machine Learning Methods

Unsupervised learning techniques have also been applied to detect abnormal transactions. Algorithms cluster unlabelled data into groups with specific patterns or characteristics to identify anomalies before ML/TF labels are studied and presented (Jiang et al., 2020). For instance, in 2016, a paper revealed unsupervised learning was employed to categorise clients into risk clusters and derive classification rules accordingly (Alexandre & Balsa, 2016). Similarly, clustering and multidimensional scaling projection were used to detect suspect groups by labelling transactional patterns well outside the normal (Sudjianto et al., 2010). Such approaches leverage aspects available within rule-based systems, such as average transaction values and structuring ratios, to enhance anomaly detection.

2.1.3 Application of Supervised Machine Learning Methods

Supervised algorithms learn to classify or predict based on labelled examples from a training dataset (Burkart & Huber, 2021). As the literature review shows, supervised machine learning algorithms are more commonly utilised by researchers to find patterns that distinguish between suspicious ML/TF and legitimate transactions based on data for which the result is known. A stochastic approximation and D-optimal design-based sequential design method was proposed to select accounts for audit (Deng et al., 2009). Similarly, neural networks and fuzzy logic achieved a 96% accuracy rate in detecting suspicious accounts (Heidarinia et al., 2014). A system that could detect group behaviour by combining network analysis and supervised learning methods, such as support vector machines and random forests, was presented (Savage et al., 2016). The system effectively detected suspicious transactions at a low false positive rate.

In recent work, a supervised machine learning model has been trained to rank financial transactions for priority investigation for their likelihood of being involved in money laundering (Jullum et al., 2020). The model learned to predict the probability that a particular transaction would be reported based on features including the sender's and receiver's background, previous behaviour, and transactions. The study found that the XGBoost framework performed well with big data, demonstrating its capability in financial crime detection. While literature is scarce, current research offers various methods and algorithms for identifying suspicious ML/TF transactions. The following section provides an extensive overview of various supervised machine-learning algorithms applied in this area.

2.2 Supervised Learning Methods

Identification of suspicious ML/TF transactions is framed as a binary classification task where the transactions are classified as either "Yes" (suspicious) or "No" (not suspicious). Some machine-learning approaches are contemplated in deciding which algorithm would best suit the given dataset. Six supervised machine learning models are selected based on their performance and popularity in binary classification tasks: K-Nearest Neighbors

(KNN), Decision Tree Classifier, Random Forest, Logistic Regression, Support Vector Machine (SVM), and Naïve Bayes Classifier. The following sections include an extensive literature review of the selected machine learning models and their ML/TF detection applications.

2.2.1 K-Nearest Neighbour (KNN)

K-Nearest Neighbour (KNN) is a classification algorithm that predicts new inputs into a category based on the similarity between the data points in the training set and new inputs. Owing to its transparency, simplicity, and robustness to noisy data, KNN is widely used for ease of use (Soofi & Awan, 2017). This non-intense learning technique, or local classification, keeps all training samples and classifies new inputs as needed. Unlike eager learning algorithms such as Neural Networks and Naïve Bayes, KNN requires minimal training time, especially when sufficient training data is available, and hence is best suited for application in scenarios with multiple class labels (Jadhav & Channe, 2013). The algorithm puts a new sample into the class of its nearest "k" neighbours, which is the majority class among them. For regression tasks, KNN approximates the result by averaging the values of the nearest neighbours.

2.2.2 Decision Tree

Decision Tree is a supervised machine learning algorithm that constructs a hierarchical tree from labelled training data to classify new instances based on their features. Nodes in the tree represent the features, possible feature values are represented by branches, and classification outcomes are represented by leaves. As a sample moves from the root to a leaf, it is tagged based on the features it encounters along the path in a divide-and-conquer fashion (Osisanwo et al., 2017). Decision Trees are classification and regression nonparametric techniques that convert data into a collection of simple if-then rules (Chong et al., 2023). The algorithm recursively divides the data in a greedy top-down manner, starting at the root and moving through internal nodes until reaching a leaf, where the final classification is made. However, Decision Trees are susceptible to

overfitting and classification error, which can be minimised by pruning techniques such as pre-pruning and post-pruning to enhance prediction accuracy (Bhavsar & Ganatra, 2012). Some well-known Decision Tree algorithms are ID3, its optimised version C4.5, and its Java implementation J48, which are well-liked for tree construction. These algorithms handle both nominal and numeric attributes with the capability of handling missing or noisy data. The CART algorithm also employs the Gini index to select attributes, with binary splits enforced and the cost-complexity pruning model applied for greater precision (Sarmah & Sarma, 2016).

2.2.3 Random Forest

Random Forest is an ensemble method composed of numerous decision trees that act collectively. It is a nonparametric, highly versatile model that can be applied to classification and regression tasks. For classification, it predicts the class by voting based on the majority of trees, and for regression, it calculates the average of all the trees' predictions (Breiman, 2001). Random Forest relies on the low correlation between its trees because non-correlated models improve ensemble predictions compared to individual tree predictions (Gregorutti et al., 2017). To achieve this, Random Forest employs two crucial techniques: bagging and feature randomness (Dietterich, 2000). Bagging, which is also known as Bootstrap Aggregating, includes sampling subsets of the training data with replacement, fitting models to these subsets, and aggregating their predictions. Because individual examples may appear multiple times across different subsets, bagging increases model stability and reduces the risk of overfitting. A common implementation of this approach is tree bagging, where decision trees are trained on bootstrap samples of the data.

Feature Randomness is at tree-building time when each tree selects a random subset of features rather than the entire feature space. This boosts model diversity, lowers tree correlation, and improves global prediction stability (Hu et al., 2021). Random Forest achieves good accuracy and predictive generalisation through training trees over different data subsets with unique sets of features. A number of tools, such as the R

package and Scikit-learn, provide means for the implementation and training of Random Forest models.

2.2.4 Logistic Regression

Logistic regression is a predictive and classification statistical method that forecasts the likelihood of an event's occurrence based on independent variables in a dataset (Nhu et al., 2020). Since the outcome will be a probability, the dependent variable is restricted between 0 and 1, where the logit transformation of the odds ratio is applied in order to estimate the predictors' and outcomes' relationship (Plakandaras et al., 2022). Being a discriminative model, Logistic Regression effectively classifies or separates classes. It can be prone to overfitting, especially when the number of predictor variables is high. In high-dimensional data, regularisation techniques are likely to prevent large coefficients and improve model generalisation. Libraries such as Scikit-learn offer efficient APIs for implementing and training Logistic Regression models. Besides classification, Logistic Regression is also widely used for anomaly detection and, hence, is a useful technique for fraud detection, particularly in banks and other financial institutions. SaaS-based businesses are increasingly using their expertise to enhance fraud prevention and protect customer interests.

2.2.5 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a classification algorithm designed for two-group classification issues. After being trained with labelled data, it comes up with correct classifications for new data points. Compared to neural networks, SVM gives greater speed and performance when working with small samples. SVM does so by designing a hyperplane that has a maximum margin between the data points of two classes. This separation ensures partitioning with distinct boundaries, improving classification accuracy and robustness (Cervantes et al., 2020). Due to its reliability and flexibility, SVM has been employed in numerous applications in various domains, particularly in pattern recognition research, where it has made significant contributions. In addition to

common classification, SVM has been transformed to solve issues like high-dimensional data, multi-class classification, and class imbalance. In addition, the integration of SVM with evolutionary algorithms and other advanced optimisation techniques has further enhanced its classification capability to make it a core tool in scientific and engineering applications (Cervantes et al., 2020).

2.2.6 Naïve Bayes

Naïve Bayes is a probabilistic classifier that is intuitive, easy to use, and resilient in performance across a broad range of tasks. It is founded on Bayes' theorem and relies on the assumption of conditional independence. It assumes each feature is independent of others given the class label (S. Chen et al., 2019). While this makes computation easier and enables efficient learning, it is never the case in practice, where there are generally complicated dependencies between features. Such constraints will influence classification accuracy, particularly in cases when there are strong inter-feature interactions. Various types of boosting solutions, i.e., attribute weighing and instance weighing, have emerged as ways to overcome such inadequacy. However, hardly any method provides complete remedies both ways simultaneously and poses challenges with handling attribute correlations (Zhang et al., 2021).

3.0 Methodology

The data for this research, obtained from Kaggle, is credit card transactions by European cardholders between September 2013. The data covers two days and comprises 283,726 transactions, of which 473 are tagged as suspicious. The data is mostly numerical variables and has been used through Principal Component Analysis (PCA) to minimise dimensionality while preserving essential information (Li & Qin, 2024). The flowchart of this work is shown in Figure 1.

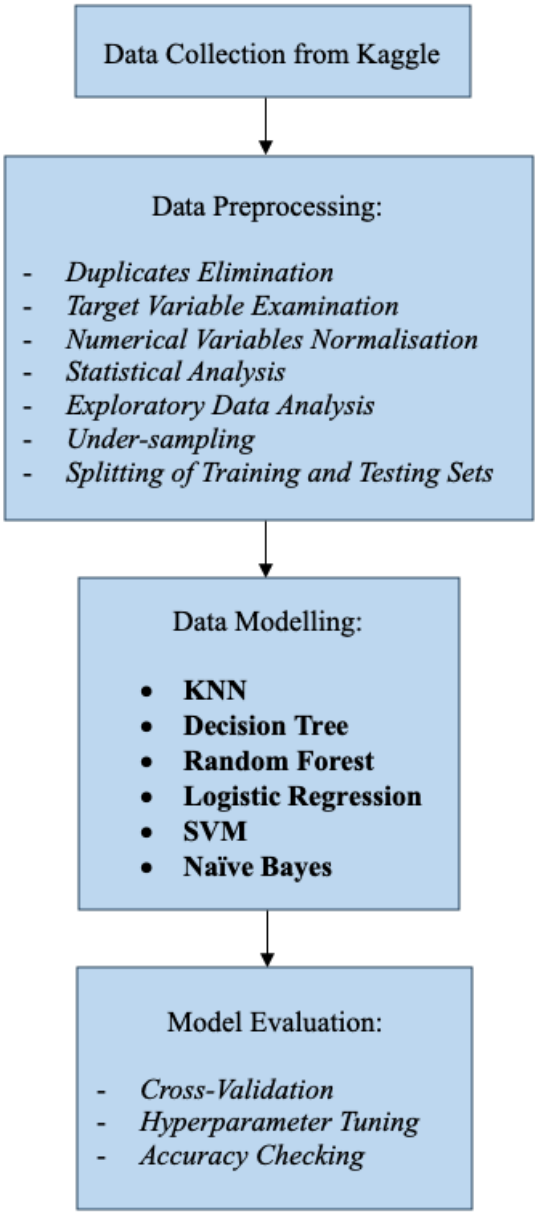

Figure 1: Methodology Flowchart

3.1 Data Preprocessing

Data preprocessing was required to prepare the dataset for use in machine learning algorithms. The dataset was initially scanned for duplicate records, and 1,081 duplicate rows were eliminated. The target variable for suspicious and legitimate transactions was then examined using the Group By function and found to be extremely imbalanced, with

473 suspicious and 283,253 legitimate transactions. 'Amount' and 'Time' variables were plotted and normalised to standardise their values. Furthermore, suspicious and legitimate transactions were separated for statistical analysis, and a sample dataset was created to portray their distribution.

The class attribute separates legitimate transactions (0) from suspicious transactions that require further investigation (1). Exploratory data analysis revealed there was an extreme class imbalance, with legitimate transactions far exceeding suspicious transactions. Four hundred seventy-three suspicious transactions and 283,253 legitimate transactions existed prior to data preparation, meaning just 0.17% of all transactions were suspicious. This lack of balance can negatively impact the performance of machine learning algorithms, as models tend to perform better when class distributions are closer to equal. In order to combat this, resampling techniques were used. Under-sampling was used in this study, removing observations of the prevailing class to achieve a more balanced set. Next, the dataset was split into training (80%), and testing (20%) sets via the train_test_split function from Scikit-learn. Machine learning models were trained on the training set, and their performance and generalisation ability were evaluated on the testing set. This strict division makes the evaluation process more credible and robust.

3.2 Data Modelling

Six models for classification were built during this phase of the research, i.e., KNN, Decision Tree, Random Forest, Logistic Regression, SVM, and Naïve Bayes. These models were then trained using the data sample that was built during the preprocessing phase. For the KNN model, the KNeighborsClassifier from sklearn.neighbors were applied, with k=2, the uncertainty of two standard deviations, or roughly a 95% confidence level. The Decision Tree model was applied using the DecisionTreeClassifier from the sklearn.tree module, with default parameters, such as the Gini impurity criterion (criterion='gini') and a minimum sample split of 2 (min_samples_split=2). Similarly, the Random Forest model was built using the RandomForestClassifier from the sklearn.ensemble module. No parameters were explicitly set, and the model size was

controlled through the max_samples parameter while utilising the default bootstrap function (bootstrap=True). For the Logistic Regression model, the LogisticRegression class from the sklearn.linear_model module was utilised with default parameters. The SVM model was implemented using the SVC class from the sklearn.svm module was also used with default parameters. Lastly, the Naïve Bayes model was implemented using the GaussianNB class from the sklearn.naive_bayes module with default parameters. These classification models serve as the foundation for the next analysis and performance testing in detecting suspicious transactions.

3.3 Model Evaluation

Three major experiments were carried out during the model evaluation process: cross-validation, hyperparameter tuning, and accuracy checking. K-Fold Cross Validation assessed classifier performance and ensured that models generalised well to new, unseen data. The training data was divided into five equally sized subsets, and each subset was used as a validation set once, while the other four were used for training. This ensured that every data point was tested, and the final cross-validation score was computed as the mean accuracy across all iterations. Hyperparameter tuning was carried out using GridSearchCV for model performance tuning. The function systematically attempts to determine pre-defined parameter values, such as the number of estimators, max depth, criterion, features, and class weight. Iterating through these combinations, fitting the model to the training set, and evaluating performance, GridSearchCV identifies the best parameter setting.

Model performance was also assessed using a confusion matrix, which categorises true positives, false positives, true negatives, and false negatives (Narkhede, 2018). This allowed for a more nuanced analysis of classification performance. Precision and recall scores were also determined, offering insight into the model's ability to refrain from false positives and false negatives respectively. Precision is the proportion of true positives to all predicted positives, while recall is the proportion of true positives to all actual positive instances.

$$Precision = \frac{True\ Positive}{True\ Positives + False\ Positives} \quad (1)$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \quad (2)$$

These performance measures provide an important insight into the model's classification performance. Figure 2 also provides a graphical representation of the Confusion Matrix to facilitate interpretability (Narkhede, 2018).

		Actual Class	
		Positive (**P**)	Negative (**N**)
Predicted Class	Positive (**P**)	True Positive (**TP**)	False Positive (**FP**)
	Negative (**N**)	False Negative (**FN**)	True Negative (**TN**)

Figure 2: Confusion Matrix

4.0 Results and Discussion

This section provides the performance results for various models experimented with, followed by identifying optimal parameters. Subsequently, a sequence of performance analyses was carried out to ascertain the accuracy of the selected model. K-Fold cross-validation was utilised as a benchmark to evaluate the performance of the KNN, Decision Tree, Random Forest, Logistic Regression, SVM, and Naïve Bayes classifiers. Figure 3 illustrates the accuracy of the testing outcome.

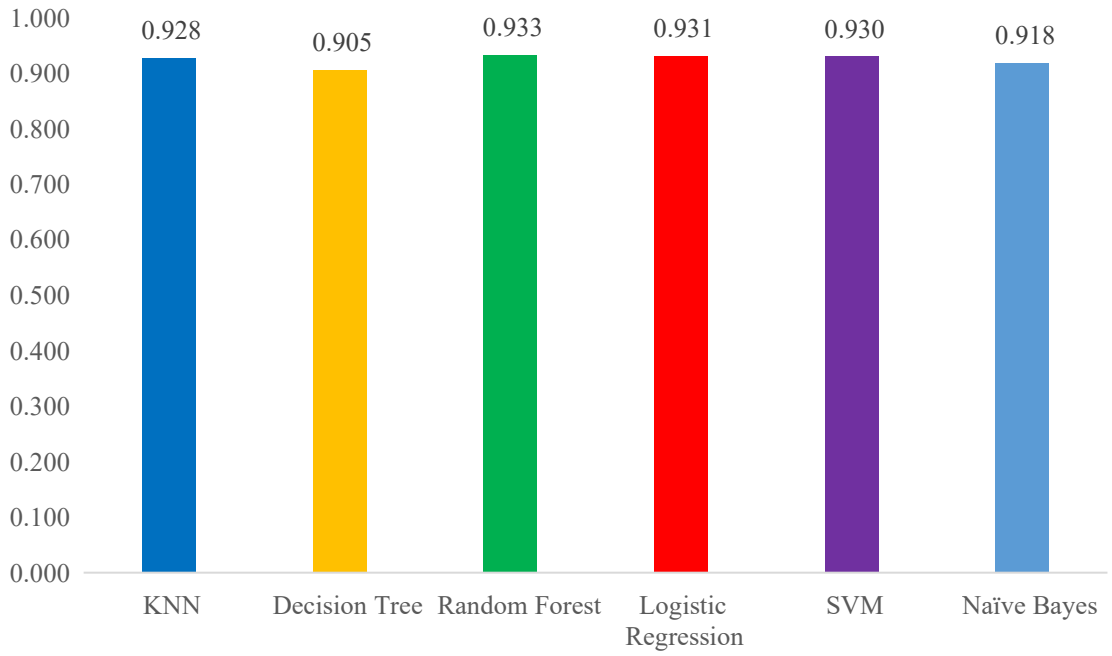

Figure 3: The Accuracy of the Supervised Machine Learning Models

Of the classification models to be tested, all of them achieved a score above 0.9, which is a good performance across the board. All the cross-validation results confirm that all the supervised machine learning algorithms performed very well. However, for its highest performance score and in order to give very accurate yet interpretable predictions, Random Forest was selected as the final model. Additionally, Random Forest can comfortably handle large data sets with many variables and process them efficiently; thus, it is the most suitable for this use.

4.1 Champion Model: Random Forest

The GridSearchCV function was employed to determine the optimal parameters for the Random Forest model. The results are presented in Table 1.

Table 1: Best Parameters for Random Forest Model

Technique	Parameters	Best Parameters
Class Weight	Balanced, Balanced_subsample	Balanced
Criterion	Gini, Entropy, Logloss	Entropy
Max_depth	2, 4, 6 8	8
Max_features	Sqrt, Log2, None	Sqrt
N_estimators	150, 200, 300, 350	350

After training the Random Forest model with the optimal parameters, the highest achieved score was 0.943. Subsequently, an accuracy evaluation yielded an accuracy score of 0.943. A confusion matrix was used to evaluate the performance of the Random Forest model, as shown in Figure 4. According to the confusion matrix, 98 transactions were correctly predicted as legitimate, while 85 were identified as suspicious. Additionally, 10 transactions were misclassified as suspicious when they were legitimate, and none of the transactions predicted as legitimate were suspicious. Based on these results, the precision score for the Random Forest model is 1.0, while the recall score is 0.895.

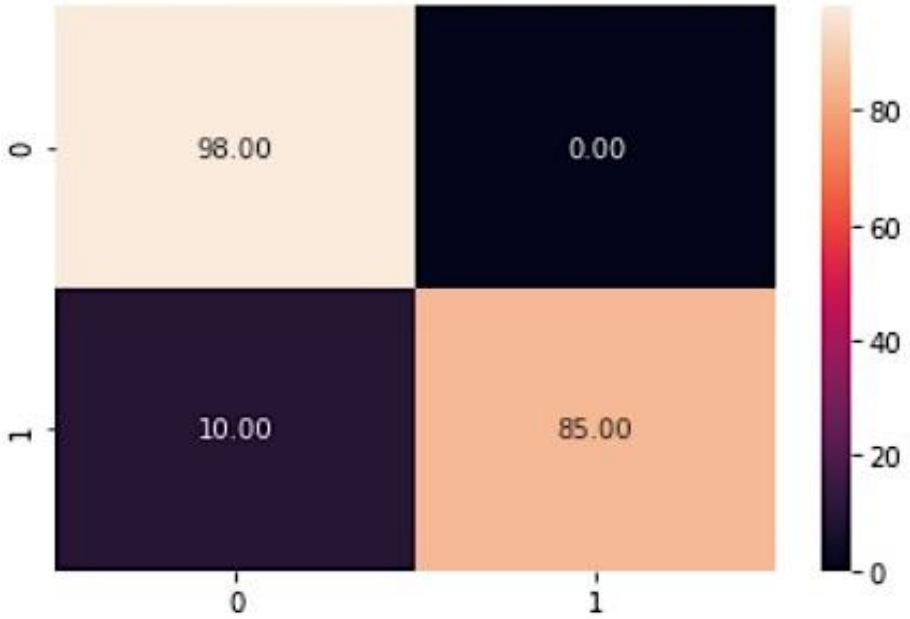

Figure 4: Confusion Matrix Result (Under-Sampling Method)

The bootstrapping resampling method was used instead of under-sampling in an alternative approach. Bootstrapping is a sampling technique that selects samples with replacement, allowing the learning algorithm to be trained on 492 samples. The confusion matrix in Figure 5 illustrates the distribution of true positives, true negatives, false positives, and false negatives resulting from this method. Specifically, 97 transactions were correctly predicted as legitimate, while 79 transactions were correctly identified as suspicious. However, 16 transactions were misclassified as suspicious when they were legitimate, and one transaction was incorrectly labelled as legitimate when it was suspicious.

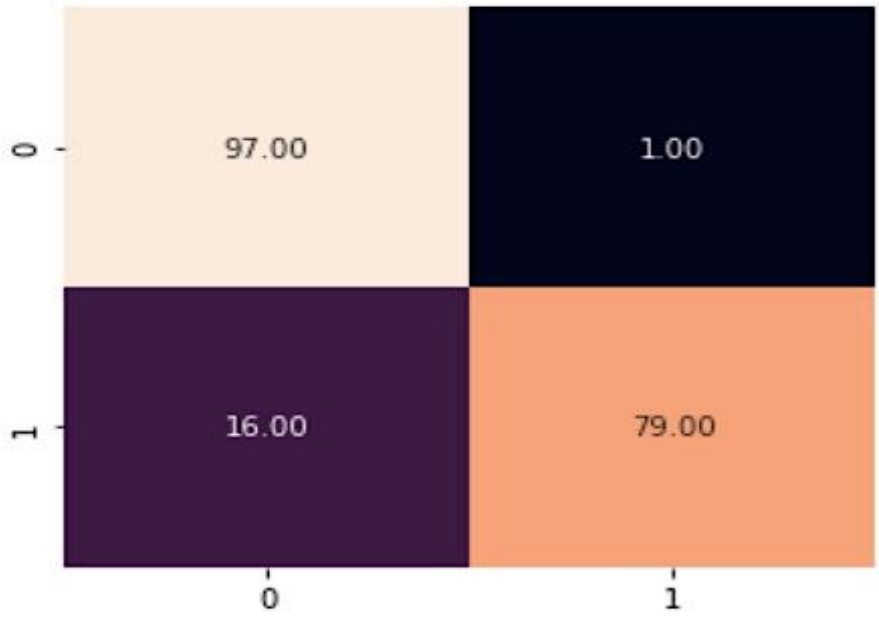

Figure 5: Confusion Matrix Result (Bootstrapping Resampling Method)

Compared to the under-sampling approach, bootstrapping resulted in decreased true positive and true negative predictions and increased false positives and false negatives. Overall, the under-sampling approach was better at balancing the dataset, and this was reflected in improved model accuracy and the elimination of false positives. Precision and recall measures are considered when choosing the best model, as they are two of the most crucial criteria for assessing a model's accuracy. A model's recall is its

capacity to locate every pertinent instance in a data collection, whereas its accuracy is its capacity to locate only pertinent data points. When it comes to unbalanced classification problems, which entail skewed data distributions because too many data points fall into one class, they work well together (Koehrsen, 2024).

5.0 Discussion

Random Forest was the optimal option due to several significant benefits. The outstanding performance of Random Forest is also reported in a paper discussing anti-money laundering advancements with AI/ML insights (Gandhi et al., 2024). In the paper, the Random Forest classifier demonstrated outstanding performance in state prediction, achieving an average accuracy of 99.99% across all states. This result highlights the model's robustness and high effectiveness in accurate state classification (Gandhi et al., 2024). Moreover, Random Forest's ability to handle imbalanced data is particularly noteworthy as it can adapt class weights to prevent misclassification of the minority class. By leveraging a combination of sampling techniques and ensemble learning, Random Forest improves dataset balance, which increases overall model performance. Its ensemble of decision trees also minimises the risk of overfitting and decreases errors in calculations, leading to more accurate predictions. Despite these strengths, under-sampling was still employed to further strengthen the model's ability to handle class imbalance effectively.

Figure 6 illustrates how the Random Forest algorithm can be integrated into the AML/CFT process. Financial institutions can use the model to detect suspicious transactions more efficiently, enabling easier manual reviews and potential escalations of flagged activity. As new suspicious transactions are uncovered, new data can be fed into the algorithm, enabling continuous learning and optimisation. This reinforcement learning process allows the model to learn over time, improving its performance in detecting new patterns of illegal financial activities. By incorporating machine learning models like Random Forest, financial institutions can automate processes, reduce false positives, and gain a deeper insight into suspicious transactions. This ultimately renders

AML/CFT programs more effective by enabling financial crimes to be detected and prevented promptly.

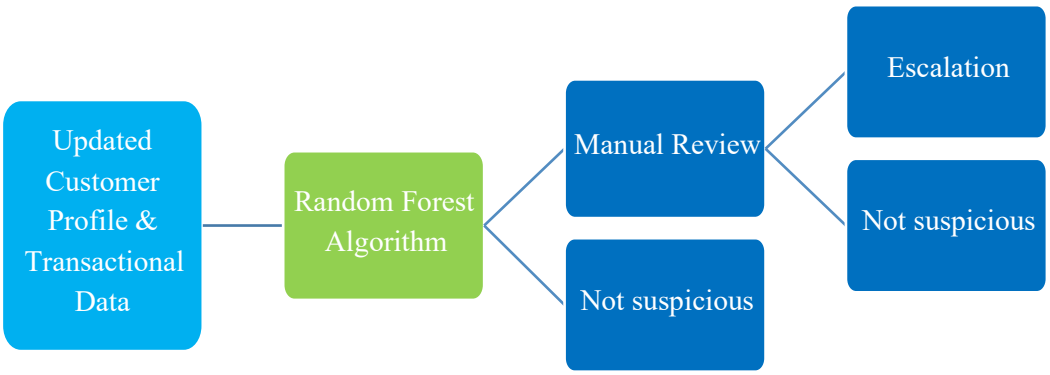

Figure 6: Reinforced Learning for Suspicious Transactions

6.0 Conclusion and Future Research

This study aimed to construct a model for predicting fraudulent transactions from vast financial data collections. Upon inspection of various supervised machine learning models such as KNN, Decision Tree, Random Forest, Logistic Regression, SVM, and Naïve Bayes, it was found that Random Forest worked best as an algorithm since it is strong and efficient in processing imbalanced datasets With the help of this model, banking institutions will have a better indication of legitimate as well as fraud transactions. Applying the Random Forest algorithm in reviewing customers' profiles and information makes the process more effective and efficient than manually processing each customer's profile. Moreover, this model can be applied to a new customer's profile as well as it is easy to run and interpret. Preprocessing had the greatest impact on model performance. Normalising the Amount and Time features achieved standardisation of the data. Under-sampling addressed class imbalance by providing a more balanced representation to be modelled. The model's performance was confirmed through performance metrics such as the confusion matrix, precision score, and recall score, with an optimal precision score of 1.0 and a high recall score of 0.874.

The results of this study may serve as a guide for practitioners to predict suspicious transactions in financial institutions based on previous patterns of transactions. Moreover, using predictive methods to detect suspicious activity may help financial institutions reduce compliance costs, which are typically higher than those of standard rule-based systems. However, the most significant limitation of this research is the invisibility of features V1 to V28 in the data because they are concealed due to confidentiality considerations. Nevertheless, the FATF, as the international standard-setting body for AML/CFT, promotes responsible technology innovation to help facilitate the practical implementation of the measures. Besides, according to the confusion matrix, the suspected transaction forecasts are mostly accurate, and no false positives have been obtained. Future research could focus on further reducing false negatives to enhance the model's real-world applicability. This study highlights the importance of dynamic and effective machine learning techniques in detecting suspicious transactions, providing valuable insights for strengthening AML/CFT processes in financial institutions.

Author Contributions Statement: The authors worked together on this paper. Conceptualisation: L.S.T., K.R.Q.; Literature Review: L.S.T., K.R.Q.; Methodology: L.S.T., K.R.Q.; Formal analysis and investigation: K.R.Q., K.K.W.; Writing – original draft preparation: L.S.T., K.R.Q.; Writing – review and editing: L.S.T., C.X.Y.; Supervision: K.K.W., C.X.Y. All authors have read and agreed to the published version of the manuscript.

Funding Statement: No funding was received to assist with the preparation of this manuscript.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data are available from the corresponding author upon request.

Acknowledgement: Special thanks were extended to the School of Management, Universiti Sains Malaysia.

Conflict of Interest Statement: The authors declare that there is no conflict of interest regarding the paper's publication.

References

Alexandre, C., & Balsa, J. (2016). *Integrating Client Profiling in an Anti-Money Laundering Multi-Agent Based System*. https://doi.org/10.1007/978-3-319-31232-3_88

Bank Negara Malaysia. (2001). *LAWS OF MALAYSIA Anti-Money Laundering, Anti-Terrorism Financing and Proceeds of Unlawful Activities Act 2001 Date of Royal Assent*.

Bank Negara Malaysia. (2019, December 31). *Malaysia's AML/CFT Regime*.

Bhavsar, H., & Ganatra, A. (2012). A Comparative Study of Training Algorithms for Supervised Machine Learning. *International Journal of Soft Computing and Engineering (IJSCE)*, *2*.

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Burkart, N., & Huber, M. F. (2021). A Survey on the Explainability of Supervised Machine Learning. In *Journal of Artificial Intelligence Research* (Vol. 70).

Cervantes, J., García-Lamont, F., Rodríguez, L., & Lopez-Chau, A. (2020). A comprehensive survey on support vector machine classification: Applications, challenges and trends. *Neurocomputing*, *408*. https://doi.org/10.1016/j.neucom.2019.10.118

Chen, S., Webb, G., Liu, L., & Ma, X. (2019). A novel selective naïve Bayes algorithm. *Knowledge-Based Systems*, *192*, 105361. https://doi.org/10.1016/j.knosys.2019.105361

Chen, Z., Le, D. V.-K., Teoh, E., Nazir, A., Karuppiah, E., & Lam, K. (2018). Machine learning techniques for anti-money laundering (AML) solutions in suspicious transaction detection: a review. *Knowledge and Information Systems*, *57*. https://doi.org/10.1007/s10115-017-1144-z

Chong, A. Y. W., Khaw, K. W., Yeong, W. C., & Chuah, W. X. (2023). Customer Churn Prediction of Telecom Company Using Machine Learning Algorithms. *Journal of*

Soft Computing and Data Mining, *4*(2), 1–22. https://doi.org/10.30880/jscdm.2023.04.02.001

Deng, X., Joseph, V. R., Sudjianto, A., & Wu, C.-F. (2009). Active Learning Through Sequential Design, With Applications to Detection of Money Laundering. *Journal of the American Statistical Association*, *104*, 969–981. https://doi.org/10.1198/jasa.2009.ap07625

Dietterich, T. (2000). An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization. *Mach. Learn.*, *40*. https://doi.org/10.1023/A:1007607513941

Estabrooks, A., Jo, D. T., & Japkowicz, N. (2004). A Multiple Resampling Method for Learning from Imbalanced Data Sets. *Computational Intelligence*, *20*, 18–36. https://doi.org/10.1111/j.0824-7935.2004.t01-1-00228.x

Financial Action Task Force. (2022, March). *FATF Recommendations*. Financial Action Task Force. https://www.fatfgafi.org/publications/fatfrecommendations/documents/fatf-recommendations.html

Gandhi, H., Tandon, K., Gite, S., Pradhan, B., & Alamri, A. (2024). Navigating the Complexity of Money Laundering: Anti-money Laundering Advancements with AI/ML Insights. *International Journal on Smart Sensing and Intelligent Systems*, *17*(1). https://doi.org/10.2478/ijssis-2024-0024

Gao, Z., & Ye, M. (2007). A framework for data mining-based anti-money laundering research. *Journal of Money Laundering Control*, *10*. https://doi.org/10.1108/13685200710746875

Gregorutti, B., Michel, B., & Saint-Pierre, P. (2017). Correlation and variable importance in random forests. *Statistics and Computing*, *27*(3), 659–678. https://doi.org/10.1007/s11222-016-9646-1

Grint, R., O'Driscoll, C., & Paton, S. (2017). *New Technologies and Anti-Money Laundering Compliance.* https://www.fca.org.uk/publication/research/new-technologies-in-aml-final-report.pdf

Heidarinia, N., Harounabadi, A., & Sadeghzadeh, M. (2014). An intelligent anti-money laundering method for detecting risky users in the banking systems. *International Journal of Computer Applications*, *97*, 35–39. https://doi.org/10.5120/17141-7780

Hu, L., Chen, J., Vaughan, J., Aramideh, S., Yang, H., Wang, K., Sudjianto, A., & Nair, V. N. (2021). Supervised machine learning techniques: an overview with applications to banking. *International Statistical Review*, *89*(3), 573–604. https://doi.org/10.1111/insr.12448

IMF Staff. (2023). *IMF policy paper 2023 review of the fund's anti-money laundering and combating the financing of terrorism strategy*. http://www.imf.org/external/pp/ppindex.aspx

Jadhav, S. D., & Channe, H. P. (2013). Comparative Study of K-NN, Naive Bayes and Decision Tree Classification Techniques. *International Journal of Science and Research (IJSR) ISSN*, *5*, 1842–1845. www.ijsr.net

Jiang, T., Gradus, J., & Rosellini, A. (2020). Supervised Machine Learning: A Brief Primer. *Behavior Therapy*, *51*. https://doi.org/10.1016/j.beth.2020.05.002

Jullum, M., Løland, A., Huseby, R., Ånonsen, G., & Lorentzen, J. (2020). Detecting money laundering transactions with machine learning. *Journal of Money Laundering Control*, *ahead-of-print*. https://doi.org/10.1108/JMLC-07-2019-0055

Koehrsen, W. (2024). *Precision and Recall: How to evaluate your classification model*. https://builtin.com/data-science/precision-and-recall#:~:text=In%20machine%20learning%2C%20precision%20and,falling%20into%20a%20single%20class.

Li, G., & Qin, Y. (2024). An Exploration of the Application of Principal Component Analysis in Big Data Processing. *Applied Mathematics and Nonlinear Sciences*, *9*. https://doi.org/10.2478/amns-2024-0664

Narkhede, S. (2018, May 9). *Understanding Confusion Matrix*. Towards Data Science. https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

Ngai, E. W. T., Hu, Y., Wong, Y. H., Chen, Y., & Sun, X. (2011). The application of data mining techniques in financial fraud detection: A classification framework and

an academic review of literature. *Decision Support Systems*, *50*(3), 559–569. https://doi.org/https://doi.org/10.1016/j.dss.2010.08.006

Nhu, V.-H., Shirzadi, A., Shahabi, H., Singh, S. K., Al-Ansari, N., Clague, J. J., Jaafari, A., Chen, W., Miraki, S., Dou, J., Luu, C., Górski, K., Thai Pham, B., Nguyen, H. D., & Ahmad, B. Bin. (2020). Shallow landslide susceptibility mapping: a comparison between logistic model tree, logistic regression, naïve bayes tree, artificial neural network, and support vector machine algorithms. *International Journal of Environmental Research and Public Health*, *17*(8). https://doi.org/10.3390/ijerph17082749

Omoseebi, A., Ola, G., & Tyler, J. (2025). *Rule-Based Systems in AML*.

Osisanwo, F. Y., Akinsola, J. E. T., Awodele O., Hinmikaiye, J. O., Olakanmi, O., & Akinjobi, J. (2017). Supervised Machine Learning Algorithms: Classification and Comparison. *International Journal of Computer Trends and Technology*, *48*, 128–138. http://www.ijcttjournal.org

Plakandaras, B., Gogas, P., Papadimitriou, T., & Tsamardinos, I. (2022). Credit Card Fraud Detection with Automated Machine Learning Systems. *Applied Artificial Intelligence*, *36*. https://doi.org/10.1080/08839514.2022.2086354

Ramentol, E., Caballero, Y., Bello, R., & Herrera, F. (2011). SMOTE-RSB *: A hybrid preprocessing approach based on oversampling and undersampling for high imbalanced datasets using SMOTE and rough sets theory. *Knowledge and Information Systems*, *33*. https://doi.org/10.1007/s10115-011-0465-6

Sarmah, J., & Sarma, S. (2016). Decision Tree based Supervised Word Sense Disambiguation for Assamese. *International Journal of Computer Applications*, *141*, 42–48. https://doi.org/10.5120/ijca2016909488

Savage, D., Wang, Q., Chou, P., Zhang, X., & Yu, X. (2016). *Detection of money laundering groups using supervised learning in networks*. https://doi.org/10.48550/arXiv.1608.00708

Soofi, A., & Awan, A. (2017). Classification Techniques in Machine Learning: Applications and Issues. *Journal of Basic & Applied Sciences*, *13*, 459–465. https://doi.org/10.6000/1927-5129.2017.13.76

Statista. (2021). *Number of online transactions in Malaysia 2020, by payment method*. https://www.statista.com/statistics/907694/malaysia-online-transaction-number-by-payment-method/

Sudjianto, A., Nair, S., Yuan, M., Zhang, A., Kern, D. A., & Cela-Díaz, F. (2010). Statistical Methods for Fighting Financial Crimes. *Technometrics*, *52*, 19–5. https://api.semanticscholar.org/CorpusID:34785896

Zhang, H., Jiang, L., & Yu, L. (2021). Attribute and instance weighted naive Bayes. *Pattern Recognition*, *111*, 107674. https://doi.org/10.1016/j.patcog.2020.107674