International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

A SUSPICIOUS FINANCIAL TRANSACTION DETECTION MODEL USING AUTOENCODER AND RISK-BASED APPROACH

GEETHA PRATHIBHA[1], A. VENEELA[2], A. SAGARIKA[3], CH. DIVYA SRI[4]
[1]Assistant Professor, Department Of It, Mallareddy College Of Engineering For Women
[2,3,4]Ug Scholar, Department Of It, Mallareddy College Of Engineering For Women

ABSTRACT

The detection of suspicious financial transactions has been a critical focus in the financial industry for decades. Traditionally, financial institutions employed rule-based systems for identifying potentially fraudulent activities. These systems rely on predefined thresholds and patterns, such as large transactions or frequent deposits, to flag suspicious activities. While effective to some extent, traditional systems face significant limitations. They often generate a high rate of false positives, requiring manual intervention to review flagged transactions. Additionally, these systems struggle to adapt to evolving fraud patterns, making them less effective in detecting sophisticated financial crimes. The growing complexity and volume of financial transactions in the digital era have heightened the need for advanced detection mechanisms. Traditional systems fail to address the dynamic nature of financial fraud, leading to inefficiencies in preventing financial losses. This creates a pressing need for a more adaptable, accurate, and scalable approach to detecting suspicious transactions. The lack of adaptability in traditional methods, combined with the significant financial and reputational risks posed by undetected fraud, underscores the necessity of a more robust detection framework. The goal is to enhance the ability to detect anomalous patterns in financial data with minimal false positives while maintaining efficiency and scalability. The proposed system introduces an innovative solution that leverages an autoencoder-based model combined with a risk-based assessment strategy. This approach aims to capture subtle anomalies in transaction data that deviate from normal patterns, enabling the identification of suspicious activities. The integration of a risk-based framework ensures that the model considers contextual factors, reducing false alarms and prioritizing high-risk transactions for further analysis. This system addresses the limitations of traditional methods, providing a sophisticated, adaptive, and reliable tool for combating financial fraud.

INTRODUCTION

Suspicious financial transaction detection is a key area of concern in the financial industry, particularly in combating fraud and money laundering. The concept emerged in the 20th century with the introduction of the first fraud detection systems, relying on manual checks and basic software tools. In India, the Reserve Bank of India (RBI) and financial institutions have been actively working towards improving fraud detection techniques. According to the Financial Intelligence Unit of India (FIU-IND), there was a significant increase in suspicious transaction reports in recent years. In 2020 alone, over 3.5 million suspicious transaction reports were filed, reflecting the growing challenge of financial fraud in the country. Traditional systems failed to keep pace with the complexity of financial crimes, including cyber fraud, phishing, and money laundering. In response, India has adopted various technological advancements to improve detection, including machine learning algorithms. The need for a more automated, accurate, and scalable solution has grown in the face of digital banking, e-commerce, and the increasing number of financial transactions. As the economy becomes more digitized, the financial sector must adapt to these changes by implementing more sophisticated detection models.

LITERATURE REVIEW

COUNTER TERRORISM FINANCE BY DETECTING MONEY LAUNDERING HIDDEN NETWORKS USING UNSUPERVISED MACHINE LEARNING ALGORITHM

- **Amr Ehab, Muhammed Shokry, +1 author N. Labib**
- **Published in Proceedings of the… 21 July 2020**

Today's most immediate threat to address is terrorism. Terror organizations use illegal methods to raise their fund, such as scamming banks, fraud, donation, ransom and oil. This illicit money needs be laundered to be used within legal economy through financial institutions (FI). This paper is a complementary to our previous research. And it's proposes an unsupervised machine learning technique for detecting Money Laundering hidden patterns, groups and transactions in a timely manner to counter terrorism finance. Two different algorithms were implemented and performance was measured, compared and summarized. The preliminary experimental results show the effectiveness of the proposed technique. Domain experts confirm that the proposed method has produced efficient accurate results by identifying and detecting similarities, hidden patterns, grouping across all transactions and all the suspicious accounts involved.

Deep Learning and Explainable Artificial Intelligence Techniques Applied for Detecting Money Laundering–A Critical Review

- Dattatray Vishnu Kute, B. Pradhan, +1 author A. Alamri
- Published in IEEE Access 2021

Money laundering has been a global issue for decades, which is one of the major threat for economy and society. Government, regulatory and financial institutions are combating it together in their respective capacity, however still billions of dollars in fines by authorities make the headlines in the news. High-speed internet services have enabled financial institutions to deliver better customer experience through multi-channel engagements, which has led to exponential growth in transactions and new avenues for laundering the money for fraudsters. Literature shows the usage of statistical methods, data mining and Machine Learning (ML) techniques for money laundering detection, but limited research on Deep Learning (DL) techniques, primarily due to lack of model interpretability and explainability of the decisions made. Several studies are conducted on application of ML for Anti-Money Laundering (AML), and Explainable Artificial Intelligence (XAI) techniques in general, but lacks the study on usage of DL techniques together with XAI. This paper aims to review the current state-of-the-art literature on DL together with XAI for identifying suspicious money laundering transactions and identify future research areas. Key findings of the review are, researchers have preferred variants of Convolutional Neural Networks, and

AutoEncoder; graph deep learning together with natural language processing is emerging as an important technology for AML; XAI use is not seen in AML domain; 51% ML methods used in AML are non-interpretable, 58% studies used sample of old real data; key challenges for researchers are access to recent real transaction data and scarcity of labelled training data; and data being highly imbalanced. Future research directions are, application of XAI techniques to bring-out explainability, graph deep learning using natural language processing (NLP), unsupervised and reinforcement learning to handle lack of labelled data; and joint research programs between research community and industry to benefit from domain knowledge and controlled access to data.

Monitor and Detect Suspicious Transactions With Database Forensic Analysis

- H. Khanuja, D. Adane
- Published in Journal of Database… 1 October 2018

The extensive usage of web has given rise to financially motivated illegal covert online transactions. So the digital investigators have approached databases for investigating undetected illegal transactions. The authors here have designed and developed a methodology to find the illegal financial transactions through the database logs. The objective is to monitor database transactions for detecting and reporting risk level of suspicious transactions. Initially, the process extracts SQL transactions from logs of different database systems, then transforms and loads them separately in uniform XML format which gives the transaction records and its metadata. The transaction records are processed with well-defined rules to get outliers present as suspicious transactions. This gives the initial belief of the transactions to be suspicious. The belief value of transactions is further rationalised using Dempster-Shafer's theory. This verifies the uncertainty and risk level of the suspected transactions to assure occurrences of fraud transactions.

EXISTING SYSTEM

The existing system for anti-money laundering (AML) and counter-terrorism financing (CFT) heavily relies on traditional approaches such as rule-based algorithms, transaction monitoring systems, and manual auditing. These systems identify unusual patterns by matching transaction data against predefined rules or thresholds. Organizations often employ data visualization techniques, risk-based assessments, and typologies to detect money laundering activities. Financial institutions also implement compliance measures guided by international standards such as the FATF recommendations. Additionally, machine learning (ML) and artificial intelligence (AI) methods have been integrated into some systems to automate the detection of suspicious activities. However, despite these advancements, the systems face several challenges, especially in handling evolving money laundering techniques and balancing regulatory compliance with financial inclusion.

Limitations

High False Positives: Rule-based systems often generate a high number of false positives, which lead to inefficiencies and wasted resources during investigations.

Adaptability to Evolving Techniques: Traditional systems struggle to adapt to new and sophisticated money laundering schemes, which involve complex patterns and multi-layered transactions.

Lack of Scalability: Existing systems may fail to handle large-scale transaction data efficiently, especially in the era of digital and mobile banking.

Financial Exclusion: Strict compliance with FATF standards can inadvertently lead to financial exclusion, particularly for underserved communities.

Limited Integration of AI/ML: While some systems use AI/ML, their integration remains limited, resulting in suboptimal performance for predictive modeling and anomaly detection.

Data Privacy Concerns: AML systems often face challenges related to data sharing and privacy compliance, especially across international borders.

Resource-Intensive Supervision: Risk-based supervision approaches demand significant resources for proper implementation, making it difficult for smaller institutions to comply.

Unintended Consequences: Over-regulation can deter financial inclusion and innovation, and misaligned priorities may lead to ineffective implementation of AML measures.

Disadvantages:

1. High Rate of False Positives

- Rule-based systems are often rigid and based on predefined criteria, which can result in a significant number of false positives. This generates excessive alerts, many of which are benign, leading to wasted resources and inefficiencies. Financial institutions must spend time and money investigating transactions that do not actually involve money laundering or terrorist financing, diverting attention away from truly suspicious activities.

2. Inability to Adapt to Evolving Techniques

- Traditional AML/CFT systems struggle to keep up with the constantly evolving methods used by money launderers and terrorist financiers. These criminals adapt quickly, using increasingly sophisticated methods, such as layered transactions, anonymized digital currencies, or cross-border financial flows. Traditional rule-based approaches may fail to identify new typologies, leaving systems vulnerable to emerging threats.

3. Limited Scalability

- As the volume of financial transactions grows exponentially, particularly with the rise of digital and mobile banking, traditional AML systems may become overwhelmed. Processing large-scale transaction data efficiently requires significant computational power and advanced technology. Existing systems may not scale effectively, leading to slow response times and the inability to analyze high volumes of data in real-time, ultimately hindering their effectiveness.

4. Financial Exclusion

- While AML/CFT regulations aim to prevent illicit activities, their strict enforcement can inadvertently lead to financial exclusion. For example, small or underserved communities may face challenges accessing financial services due to stringent compliance checks or a fear of triggering alerts. This can limit the availability of financial products and services for individuals and businesses in emerging markets or rural areas, exacerbating inequality.

5. Resource-Intensive Supervision

- Risk-based supervision is a valuable approach, but it requires significant resources for effective implementation, such as skilled personnel, specialized technology, and ongoing training. For smaller financial institutions or those in less-developed regions, the cost and complexity of compliance can be prohibitive. This creates a disproportionate burden on these institutions, reducing their ability to compete and innovate while ensuring compliance with international AML/CFT standards.

- These limitations highlight the need for more adaptive, scalable, and resource-efficient approaches to AML/CFT compliance, as well as the integration of AI/ML technologies to reduce manual interventions and enhance predictive capabilities**.**

PROPOSED SYSTEM Fraud detection in financial transactions is a critical task that requires a robust and systematic approach to identify anomalies and prevent financial losses. This research outlines a step-by-step methodology for developing and comparing two machine learning models: an existing Deep Neural Network (DNN) model and a proposed Autoencoder-Random Forest (AERF) model. The procedure involves dataset preparation, preprocessing, model training, and performance evaluation. Below are the detailed steps of this research.

Step 1: Upload Dataset

The first step is to uploading the dataset. The dataset used in this research is assumed to contain financial transaction data, including features such as transaction type, amount, origin and destination balances, and a target column indicating whether the transaction is fraudulent (`isFraud`).

The dataset is uploaded through a web-based interface developed using Django. This interface allows users to select and upload a CSV file. The uploaded dataset is then processed and temporarily stored for further analysis. Ensuring that the dataset is clean, structured, and appropriately formatted is crucial to the success of the subsequent steps. A robust upload mechanism checks for file type and basic data integrity to prevent errors during processing.

Step 2: Data Preprocessing

Data preprocessing is a fundamental step to ensure the dataset is suitable for machine learning models. The following preprocessing tasks are performed:

1. **Handling Null Values:** Missing data can lead to inaccurate results or model errors. A thorough analysis of null values is conducted. For columns with a significant number of missing values, strategies like imputation (mean, median, or mode) or column removal are applied based on their importance.
2. **Label Encoding:** Many machine learning algorithms require numerical input. Categorical columns, such as the

`type` of transaction, are encoded using label encoding. This transforms categorical labels into integer representations. For example, transaction types like 'CASH_IN' and 'CASH_OUT' are converted to numerical labels such as 0 and 1.

3. **Feature Selection:** Key features such as `amount`, `oldbalanceOrg`, `newbalanceOrig`, `oldbalanceDest`, and `newbalanceDest` are selected for model training. These features are chosen based on domain knowledge and their relevance to detecting fraudulent transactions. The target column, `isFraud`, serves as the label for classification tasks.
4. **Data Standardization:** To ensure that all features contribute equally to the model, standardization is applied. This scales the numerical features to have zero mean and unit variance, which improves the performance and convergence of models like DNNs and Autoencoders.

Step 3: Train-Test Splitting (80-20 Ratio)

The dataset is split into training and testing sets using an 80-20 ratio. This means 80% of the data is used for model training, while 20% is reserved for testing. This split ensures that the models can generalize well to unseen data. The `train_test_split` function from the scikit-learn library is used for this purpose, ensuring a random but reproducible division.

Care is taken to preserve the distribution of the target variable (`isFraud`) in both sets, especially since fraud datasets are often imbalanced. Techniques like stratified sampling are employed to maintain the proportion of fraudulent and non-fraudulent transactions in the training and testing sets.

Step 4: Existing DNN Model Building

A Deep Neural Network (DNN) model is implemented as the baseline for fraud detection. The architecture and training of the DNN are as follows:

1. **Model Architecture:**
   - o Input Layer: Accepts features like `amount` and balances.
   - o Hidden Layers: Two fully connected layers with 128 and 64 neurons, respectively, and ReLU activation functions.
   - o Output Layer: A softmax layer for binary classification (fraudulent or non-fraudulent).
2. **Training Configuration:**
   - o Loss Function: Categorical cross-entropy is used as the loss function, suitable for multi-class classification problems.
   - o Optimizer: The Adam optimizer is employed for its efficiency in handling sparse gradients.
   - o Metrics: Accuracy, precision, recall, and F1-score are tracked during training.
3. **Early Stopping:** Early stopping is implemented to halt training when the model's performance on the validation set stops improving, preventing overfitting.

Step 5: Proposed Autoencoder with RF Model Building

The proposed model combines the power of Autoencoders for feature extraction with the robustness of Random Forests for classification. The steps for building this model are as follows:

1. **Autoencoder for Feature Extraction:**

○ The Autoencoder is an unsupervised neural network designed to reconstruct input data.

○ Architecture: It consists of an encoder (compresses input into a lower-dimensional representation) and a decoder (reconstructs the input from the compressed features).

○ Training: The Autoencoder is trained to minimize reconstruction loss, measured by mean squared error (MSE). Once trained, the encoder part is extracted and used to transform the original features into a compact, informative representation.

2. **Random Forest Classifier:**

○ The transformed features from the Autoencoder are used as input for the Random Forest Classifier.

○ Random Forest is an ensemble learning method that combines multiple decision trees to improve classification performance and reduce overfitting.

○ Hyperparameter Tuning: Parameters like the number of trees and maximum depth are optimized to achieve the best performance.

Step 6: Performance Comparison

The performance of the DNN and AERF models is compared using standard evaluation metrics:

1. **Accuracy:** Measures the proportion of correctly classified transactions out of the total.
2. **Precision:** Indicates the proportion of true positive predictions among all positive predictions.
3. **Recall:** Represents the proportion of actual fraudulent transactions correctly identified.
4. **F1-Score:** The harmonic mean of precision and recall, providing a balanced measure of performance.
5. **Confusion Matrix:** Visualizes the number of true positives, false positives, true negatives, and false negatives.

Advantages :

1. Improved Fraud Detection Accuracy with Autoencoder-Random Forest (AERF) Model

- Advantage: The AERF model combines the strengths of Autoencoders and Random Forests, improving fraud detection by extracting compact, informative features from the transaction data (using Autoencoders) before classifying it with the Random Forest algorithm. Autoencoders are particularly good at learning data patterns without supervision, which can capture complex and subtle fraud patterns that traditional models may miss. The Random Forest classifier, being an ensemble method, effectively handles noisy data and improves classification accuracy by aggregating results from multiple decision trees.

2. Scalability and Robustness

- Advantage: The Random Forest component in the AERF model is highly scalable, meaning it can handle large datasets efficiently. Random Forest's ability to work with a large number of trees and its feature selection capabilities allow it to maintain robust performance even as the size of transaction data grows. This is particularly useful in a financial environment where transaction volumes are constantly increasing.

3. Dimensionality Reduction and Improved Feature Representation (Autoencoder)

- Advantage: The Autoencoder used in the AERF model performs dimensionality reduction, compressing the input data into a lower-dimensional representation that retains the most important features. This helps in reducing noise and focusing the model's attention on the most relevant patterns, which can improve overall performance, especially in cases where the raw data has many irrelevant or redundant features. This can also help in reducing overfitting and enhancing generalization on unseen data.

4. Handling Imbalanced Datasets Effectively

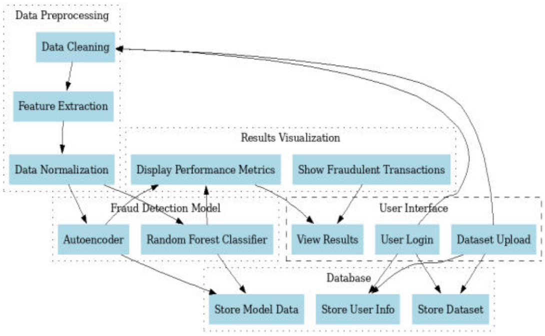
- Advantage: Fraud detection datasets are often imbalanced, with a small proportion of fraudulent transactions compared to legitimate ones. Both the DNN and AERF models are designed to handle this imbalance effectively. In particular, the AERF model can benefit from Random Forest's built-in feature importance and ensemble learning, which helps it focus on the minority class (fraudulent transactions) without being overwhelmed by the majority class (legitimate transactions). Additionally, techniques like stratified sampling during data splitting ensure that the proportion of fraudulent transactions is preserved, improving model reliability.

5. Interpretability and Transparency (Random Forest)

- Advantage: One of the challenges of deep learning models like DNNs is the "black-box" nature of their predictions, making it hard to interpret how they arrive at a decision. However, the Random Forest component in the AERF model provides greater transparency in its decision-making process. Random Forests are inherently interpretable since they aggregate decisions made by individual decision trees, allowing users to analyze feature importance and understand the reasoning behind predictions. This makes the AERF model more suitable for practical deployment in financial institutions where explainability is important for compliance and auditing purposes.

IMPLEMENTATION

SYSTEM ARCHITECTURE



MODULES

TensorFlow

TensorFlow is a free and open-source software library for dataflow and differentiable programming across a range of tasks. It is a symbolic math library and is also used for machine learning applications such as neural networks. It is used for both research and production at Google.

TensorFlow was developed by the Google Brain team for internal Google use. It was released under the Apache 2.0 open-source license on November 9, 2015.

NumPy

NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.

It is the fundamental package for scientific computing with Python. It contains various features including these important ones:

- A powerful N-dimensional array object
- Sophisticated (broadcasting) functions
- Tools for integrating C/C++ and Fortran code
- Useful linear algebra, Fourier transform, and random number capabilities

Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary datatypes can be defined using NumPy which allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

Pandas

Pandas is an open-source Python Library providing high-performance data manipulation and analysis tool using its powerful data structures. Python was majorly used for data munging and preparation. It had very little contribution towards data analysis. Pandas solved this problem. Using Pandas, we can accomplish five typical steps in the processing and analysis of data, regardless of the origin of data load, prepare, manipulate, model, and analyze. Python with Pandas is used in a wide range of fields including academic and commercial domains including finance, economics, Statistics, analytics, etc.

Matplotlib

Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and IPython shells, the Jupyter Notebook, web application servers, and four graphical user interface toolkits. Matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, error charts, scatter plots, etc., with just a few lines of code. For examples, see the sample plots and thumbnail gallery.

For simple plotting the pyplot module provides a MATLAB-like interface, particularly when combined with IPython. For the power user, you have full control of line styles, font properties, axes properties, etc, via an object-oriented interface or via a set of functions familiar to MATLAB users.

Scikit – learn

Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python. It is licensed under a permissive simplified BSD license and is distributed under many Linux distributions, encouraging academic and commercial use. Python

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace.

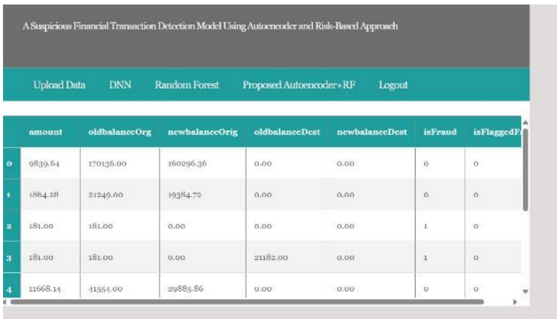
Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

- Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.

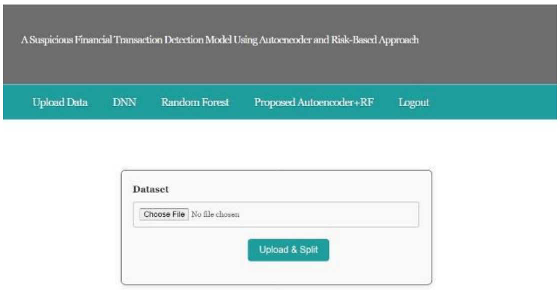
- Python is Interactive − you can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

Python also acknowledges that speed of development is important. Readable and terse code is part of this, and so is access to powerful constructs that avoid tedious repetition of code. Maintainability also ties into this may be an all but useless metric, but it does say something about how much code you have to scan, read and/or understand to troubleshoot problems or tweak behaviors. This speed of development, the ease with which a programmer of other languages can pick up basic Python skills and the huge standard library is key to another area where Python excels. All its tools have been quick to implement, saved a lot of time, and several of them have later been patched and updated by people with no Python background - without breaking

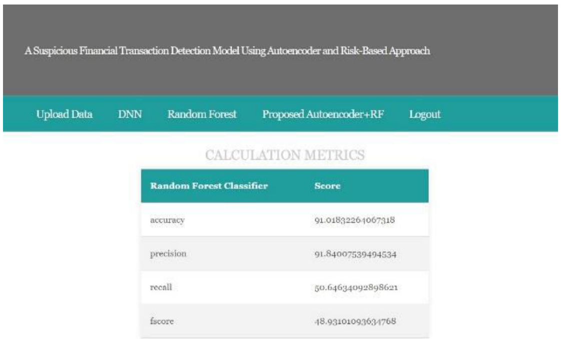
RESULT


A Suspicious Financial Transaction Detection Model Using Autoencoder and Risk-Based Approach

	amount	oldbalanceOrg	newbalanceOrig	oldbalanceDest	newbalanceDest	isFraud	isFlaggedF
0	9839.64	170136.00	160296.36	0.00	0.00	0	0
1	1864.28	21249.00	19384.72	0.00	0.00	0	0
2	181.00	181.00	0.00	0.00	0.00	1	0
3	181.00	181.00	0.00	21182.00	0.00	1	0
4	11668.14	41554.00	29885.86	0.00	0.00	0	0


A Suspicious Financial Transaction Detection Model Using Autoencoder and Risk-Based Approach

A Suspicious Financial Transaction Detection Model Using Autoencoder and Risk-Based Approach


A Suspicious Financial Transaction Detection Model Using Autoencoder and Risk-Based Approach

CALCULATION METRICS

Random Forest Classifier	Score
accuracy	91.01832264067318
precision	91.84007539494534
recall	50.64634092898621
fscore	48.93101093634768

CONCLUSION

The Research combines the strengths of **Autoencoders** for feature extraction and dimensionality reduction with the **Random Forest Classifier (RFC)** for robust classification to detect fraudulent transactions in financial datasets. This hybrid approach leverages the unsupervised learning capabilities of Autoencoders to identify hidden patterns and anomalies in transaction data while utilizing RFC's high accuracy and interpretability for classification tasks. The system's architecture ensures scalability, efficiency, and accuracy in fraud detection, addressing challenges posed by imbalanced datasets and complex transactional behaviors. The project significantly contributes to reducing financial losses and enhancing trust in financial systems.

REFERENCES

[1] Singh, K., & Best, P. (2019). Anti-money laundering: Using data visualization to identify suspicious activity. International

Journal of Accounting Information Systems, 34, 100418.

[2] Whisker, J., & Lokanan, M. E. (2019). Anti-money laundering and counter-terrorist financing threats posed by mobile money. Journal of Money Laundering Control, 22(1), 158-172.

[3] Dobrowolski, Z., & Sułkowski, Ł. (2019). Implementing a sustainable model for anti-money laundering in the United Nations development goals. Sustainability, 12(1), 244.

[4] Samantha Maitland Irwin, A., Raymond Choo, K. K., & Liu, L. (2011). An analysis of money laundering and terrorism financing typologies. Journal of Money Laundering Control, 15(1), 85-111.

[5] Uthayakumar, J., Vengattaraman, & Dhavachelvan, T. P. (2022). Swarm intelligence based classification rule induction (CRI) framework for qualitative and quantitative approach: An application of bankruptcy prediction and credit risk analysis. Journal of King Saud University - Computer and Information Sciences, 32(6), 647-657.

[6] KOFIU. (2017). Risk-Based Approach (RBA) Processing Standards for AML/CFT in Financial Investment Businesses. Institutional Operations Division, Financial Intelligence Unit.

[7] Lee, C.-J., & Lee, J.-C. (2013). Experiences and methodology of Korea's anti-money laundering system deployment and development. Knowledge Sharing Program: KSP Modularization.

[8] Pavlidis, G. (2023). The dark side of anti-money laundering: Mitigating the unintended consequences of FATF standards. Journal of Economic Criminology, 100040.

[9] Celik, K. (2021). Impact of the FATF Recommendations and their Implementation on Financial Inclusion: Insights from Mutual Evaluations and National Risk Assessments.

[10] Jayasekara, S. D. (2018). Challenges of implementing an effective risk-based supervision on anti-money laundering and countering the financing of terrorism under the 2013 FATF methodology. Journal of Money Laundering Control, 21(4), 601-615.

[11] Raghavan, K. R. (2006). Integrating anti-money laundering into the compliance structure: How the requirements for compliance with BSA/AML are changing the emphasis of corporate governance and finance functions. Bank Accounting & Finance, 19(6), 29-37.

[12] Raghavan, K. R. (2006). Integrating anti-money laundering into the compliance structure: How the requirements for compliance with BSA/AML are changing the emphasis of corporate governance and finance functions. Bank Accounting & Finance, 19(6), 29-37.

[13] Labib, N. M., Rizka, M. A., & Shokry, A. E. M. (2020). Survey of machine learning approaches of anti-money laundering techniques to counter terrorism finance. In Internet of Things—Applications and Future: Proceedings of ITAF 2019 (pp. 73-87). Springer.

[14] Cherif, A., Badhib, A., Ammar, H., Alshehri, S., Kalkatawi, M., Imine, A. (2023). Credit card fraud detection in the era of disruptive technologies: A systematic review, Journal of King Saud University - Computer and Information Sciences, 35(1), 145-174.

[15] Senator, T. E., Goldberg, H. G., Wooton, J., Cottini, M. A., Khan, A. U., Klinger, C. D., Llamas, W. M., Marrone, M. P., & Wong, R. W. (1995). Financial crimes enforcement network AI system

(FAIS) identifying potential money laundering from reports of large cash transactions. AI magazine, 16(4), 21-21.

[16] Wang, S.-N., & Yang, J.-G. (2007). A money laundering risk evaluation method based on decision tree. 2007 international conference on machine learning and cybernetics, January.

[17] Zhang, D., & Zhou, L. (2004). Discovering golden nuggets: Data mining in financial application. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 34(4), 513-522.