| Volume 02 | Issue 01 | June 2026 | ISSN: 3139-1737 |

International Journal of Advances and Applications in Convergent Studies & Transdisciplinary (IJ-AACST)

Bridging Disciplines, Bridging Knowledge

Website: https://aacstij.org | Email: journaleditor@aacstij.org

Credit Card Fraud Detection Using Machine Learning

Dr. Varsha C. Pande[1]*, Dr. Abha S. Khandelwal [2]

[1] Assistant Professor, Department of Computer Science, Shivaji Science College, Nagpur, Maharashtra, India, varshapande.var@gmail.com

https://orcid.org/0009-0001-6241-0465

[2] Retired Head, Department of Computer Science, Hislop College, Nagpur, Maharashtra, India

abha.ak@gmail.com

ABSTRACT

In the digital age, the proliferation of online services and the convenience of electronic payments have transformed consumer behavior globally. Among the most widely adopted payment methods are credit cards, which offer quick, cashless, and remote transaction capabilities. However, this digital convenience has also led to a significant rise in cybercrimes, particularly **credit card fraud**. As the volume of credit card transactions continues to grow exponentially, so does the sophistication of fraudulent schemes. Hence, the development of **robust, real-time fraud detection systems** has become a critical focus in financial technology and cyber security. In this research we implemented and compared Decision Tree, Random Forest, and Hybrid (RF+DT) models, The results demonstrate that ensemble and hybrid approaches provide more robustness and accuracy compared to single models.

KEYWORDS: Decision Tree, Hybrid (RF+DT) models, Random Forest.

1. INTRODUCTION

Credit card is a small thin plastic or fiber card that contains information about the person such as picture or signature and person named on it to charge purchases and service to his linked account charges for which will be debited regularly. Now a day's card information is read by ATM's, swiping machines, store readers, bank and online transaction. Each card as a unique card number which is very important, its security is mainly relies on physical security of the card and also privacy of the credit card number [1].

Credit card fraud typically involves unauthorized usage of credit card information for financial gain. Common methods include phishing, identity theft, and card skimming. Unlike traditional fraud detection mechanisms that depend heavily on predefined rules or manual verification, **machine learning (ML)** provides a more adaptive, data-driven approach. ML algorithms can learn transaction patterns from vast datasets and detect anomalies that deviate from typical consumer behavior. As a result, they can identify fraudulent transactions with increasing accuracy and efficiency [2].

Machine Learning (ML): Machine learning techniques allow fraud detection systems to adapt automatically by learning transaction patterns directly from historical data, eliminating the need for manually defined rules and improve their performance without being directly programmed with specific instructions. Instead of following fixed rules, machine learning models are trained using large amounts of data to recognize patterns, make predictions, or take decisions.

For example, a machine learning system can be trained to identify whether an email is spam or not, predict stock prices, recommend products, or detect fraudulent transactions. The more data the system is exposed to, the better it becomes at adapting and making accurate decisions. Machine learning is widely used today in areas like healthcare, finance, e-commerce, autonomous vehicles, and many other industries, making it one of the most important technologies in the modern world [1]. Machine Learning is generally classified into **three main types:**

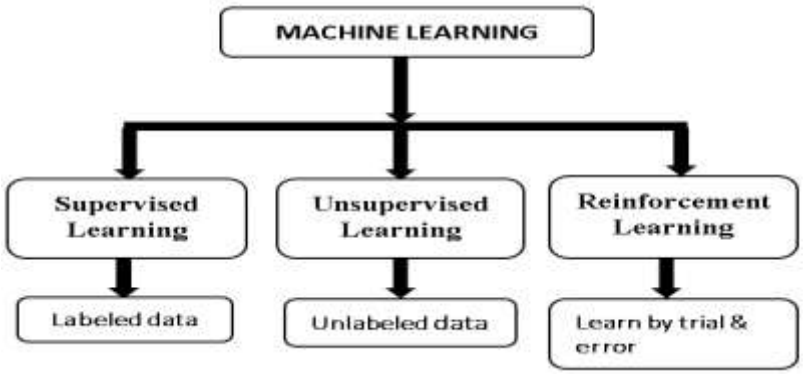


Fig 1: Classification of machine learning

- **Supervised Learning:** Supervised machine learning is a method where models are trained using labelled data (input with the correct output). It learns the relationship between inputs and outputs to make predictions on new data. For example, classifying emails as spam or not spam, or predicting house prices. It mainly includes classification (categorizing data) and regression (predicting continuous values) [2].

- **Unsupervised Learning:** Unsupervised machine learning is a type of machine learning where the model is trained using data that does not have any labels or predefined outputs. Unlike supervised learning, where the system learns from input-output pairs, unsupervised learning algorithms work on raw data and try to find hidden patterns, structures, or relationships within it. The main goal is to group similar data points together or reduce the complexity of data while retaining important information. Common techniques in unsupervised learning include clustering and visualization.

- **Reinforcement Learning:** Reinforcement Learning (RL) is a type of machine learning where an agent learns to make decisions by interacting with an environment. Instead of learning from labelled data, the agent learns by receiving rewards or penalties based on its actions. The goal is to maximize cumulative rewards over time by figuring out the best strategy, called a policy [2].

Machine Learning Algorithms: -

1. Linear Regression: - Linear Regression is a supervised learning algorithm used to predict continuous numerical values. It models the relationship between the target (dependent variable) and one or more features (independent variables) using a straight line. The algorithm tries to find the best-fitting line that minimizes the difference between predicted and actual values. [1].

2. Logistic Regression: - Logistic Regression is used for classification tasks, especially binary classification (yes/no, true/false). Instead of predicting a number, it predicts the probability of an event using the logistic (sigmoid) function. If the probability is above a certain threshold (usually 0.5), the data point is classified into one class; otherwise, into the other. It is simple, effective, and interpretable. Logistic Regression works well when the output depends on several features [1].

3. Decision Tree: - Decision Tree is a supervised algorithm used for classification and regression. It works like a flowchart, where each node asks a question about the data features, each branch represents possible answers, and leaf nodes give the final prediction. It is useful for identifying

patterns and making clear decisions. However, single trees can overfit, so ensemble methods like Random Forest are often used [1].

4. Support Vector Machines (SVM): - SVM is a supervised algorithm used for classification and regression tasks. It works by finding the best boundary (hyperplane) that separates different classes with the maximum margin. The closest points from each class that touch the boundary are called support vectors. SVM can handle linear and non-linear data using kernel functions. It performs well in high-dimensional spaces and is robust to overfitting in smaller datasets [8].

5. K-Nearest Neighbours (k-NN):- k-NN is a simple, lazy learning algorithm used for classification and regression. It predicts the output of a new data point by looking at the k closest points in the training set. The class or value is determined based on the majority (for classification) or average (for regression) of the neighbours. It is non-parametric, meaning it makes no assumptions about the data distribution [1].

Random Forest: - Random Forest is an ensemble learning method that combines multiple decision trees to make predictions. Each tree is trained on a random subset of data and features, and all trees vote for the final result (classification) or average the results (regression). This approach reduces errors, improves accuracy, and avoids overfitting [1,2].

2. LITERATURE REVIEW / RELATED WORK

Khatri et al. [1], implemented several ML algorithms for credit card fraud detection. In this research, the authors implemented the following methods: Decision Tree (DT), k-Nearest Neighbor (kNN), Logistic Regression (LR), Random Forest (RF), and Naïve Bayes (NB). To evaluate the ML-based credit card fraud detection models, the researchers used a dataset that was generated from European cardholders in 2013. Moreover, the authors considered the sensitivity and the precision as the main performance metrics. The results showed that the KNN algorithm achieved the most optimal results with a precision of 91.11% and a sensitivity of 81.19%.

Trivedi et al. [2], proposed an efficient credit card fraud detection engine using ML methods. In this research, the authors considered many supervised ML techniques including Gradient Boosting (GB) and Random Forest (RF). The authors evaluated these methods using the European cardholder's dataset. The performance metrics used to assess the effectiveness of the proposed approaches include the accuracy and the precision. The outcome of the experiments showed that the GB obtained an accuracy of 94.01% and a precision of 93.99%. On the other hand, the RF achieved an accuracy of 94.00% and a precision of 95.98%.

Riffi et al. [3], implemented a credit card fraud detection engine using the Extreme Learning Machine (ELM) and Multilayer Perceptron (MLP) algorithms. Both the ELM and MLP are artificial neural networks (ANNs); however, they differ in terms of internal architecture. In this research, the authors used the European cardholder's dataset that was generated in 2013. The authors used the fraud detection accuracy as the main performance metric. The results demonstrated that the MLP method achieved an accuracy of 97.84%. In contrast, the ELM attained credit card fraud detection accuracy of 95.46%. This work concluded that the MLP outperformed the ELM; however, the ELM is less complex in comparison to the MLP.

J. Esmaily and R. Moradinezhad [6], in their paper proposed a hybrid of artificial neural network and decision tree. In their model they used a two-phase approach. In first phase the classification results of Decision tree and Multilayer perceptron were used to generate a new dataset which in second phase is feed into Multilayer perceptron to finally classify the data. This model promises reliability by giving very low false detection rate.

Tanmay Kumar and Suvasini Panigrahi [7], in their paper proposed a hybrid approach to credit card fraud detection using fuzzy clustering and neural network. It makes use of two phases.

3. RESEARCH METHODOLOGY

The methodology for credit card fraud detection begins with **dataset collection and preprocessing**. A dataset containing legitimate and fraudulent transactions, such as the European credit card transaction dataset, is used. Preprocessing involves handling missing values, encoding categorical features, normalizing numerical values, and addressing the class imbalance problem, typically with SMOTE (Synthetic Minority Oversampling Technique).

Feature selection methods like correlation analysis, recursive feature elimination, or genetic algorithms are applied to identify the most relevant attributes for fraud detection. After preprocessing, the dataset is split into training and testing subsets, usually in an 80:20 ratio, to build and validate the model effectively.

In the next stage, **machine learning algorithms** such as Logistic Regression, Decision Trees, Random Forests, and ensemble techniques are applied to the training dataset. The models learn transaction patterns and are tested on unseen data to evaluate their predictive ability. Performance metrics like accuracy, precision, recall, F1-score, AUC-ROC curve, and confusion matrix are used for assessment.

A feedback loop involving hyper parameter tuning and feature refinement ensures improved accuracy and generalization. Once the model achieves reliable performance, it can be deployed for real-time fraud detection, providing financial institutions with an automated and efficient tool to minimize fraud while reducing false positives.

Models and Dataset:

Logistic Regression: - Logistic Regression used for classification tasks, especially when the outcome has two options, like fraud or not fraud. Instead of predicting a continuous value, it estimates the probability of a transaction being fraudulent using the logistic (sigmoid) function. If the probability is above a certain threshold (commonly 0.5), the transaction is classified as fraud; otherwise, it is considered genuine.

In credit card fraud detection, this algorithm is useful because it can analyses multiple features of a transaction, such as amount, location, time, and spending patterns, to determine whether the behaviors look suspicious. Logistic Regression is simple, fast, and effective, making it a good choice for detecting fraud in large datasets.

Decision Tree: - A Decision Tree algorithm works like a flowchart, where each internal node represents a feature or attribute, each branch represents a decision rule based on that feature, and each leaf node represents the final outcome or class. In credit card fraud detection, Decision Trees are used to classify transactions as either fraudulent or legitimate. The algorithm analyses historical transaction data, including features such as transaction amount, time, location, and cardholder behaviour, and learns rules that separate fraudulent transactions from normal ones.

Random Forest: - Random Forest is a machine learning algorithm that works by combining many decision trees to make better and more accurate predictions. Think of it like asking the opinion of a group of experts instead of just one: each tree gives its own decision, and the forest chooses the most popular answer. This approach helps reduce errors and makes the model more reliable. In credit card fraud detection, Random Forest can analyse transaction data, such as the amount, location, time, and type of purchase, to determine whether a transaction is normal or potentially fraudulent. By learning patterns from past fraudulent and genuine transactions, it can flag suspicious activity quickly, helping banks prevent fraud while keeping most legitimate transactions unaffected.

Dataset:- The dataset (**credit_card_fraud_dataset.csv**) contains **100,000 transaction records** with 7 attributes: TransactionID, TransactionDate, Amount, MerchantID, TransactionType, Location, and IsFraud. This dataset provides a more realistic representation of transaction details, such as the merchant's identity, the type of transaction (purchase or refund), the location, and the monetary amount. The target variable IsFraud indicates

whether a transaction is fraudulent (1) or genuine (0). Since it includes clear contextual information like time, merchant, and geography, this dataset is particularly useful for feature engineering and for building fraud detection models that consider behavioral and contextual patterns.

Tools and Libraries: -

In this research, we used **Jupyter Notebook** platform.

Jupyter Notebook is an open-source, interactive web-based application that allows users to write and execute Python code in a structured and organized way. It is widely used in data science, machine learning, and academic research because it integrates live code, visualizations, and explanatory text within a single document.

Libraries:

We used NumPy (Numerical Python), Pandas, Scikit-learn and Matplotlib libraries used for implementation.

The methodology typically follows a systematic approach encompassing data pre-processing, model selection, training, evaluation, and performance comparison.

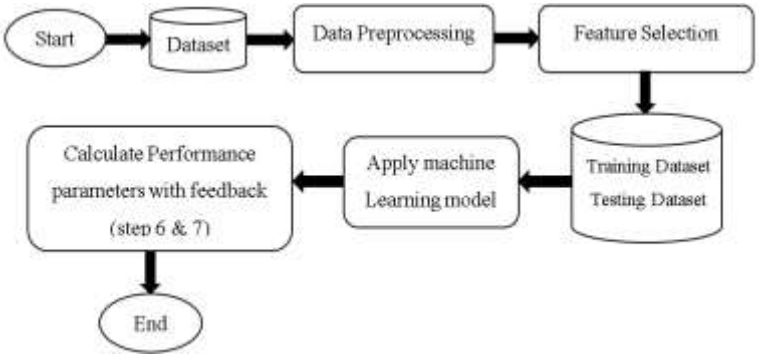


Fig 2: Steps in Credit Card Fraud Detection

- **Dataset Collection**

The process begins by acquiring a dataset that contains transaction records, including both legitimate and fraudulent entries. A commonly used dataset is the European credit card transaction dataset.

- **Data Preprocessing**

The collected dataset undergoes preprocessing to ensure data quality. This includes:

  - Handling missing or null values
  - Encoding categorical variables (if any)
  - Normalizing or standardizing numerical features
  - Dealing with class imbalance using techniques like SMOTE (Synthetic Minority Over-sampling Technique)

- **Feature Selection**

From the preprocessed data, only the most relevant features are selected. This helps improve model accuracy and reduce computational complexity. Feature selection techniques may include:

  - Correlation analysis
  - Recursive Feature Elimination (RFE)

- Genetic Algorithm or statistical methods

- **Dataset Splitting**

The dataset is then split into two subsets:

- **Training Dataset:** Used to train the machine learning model.
- **Testing Dataset:** Used to evaluate model performance.
  This is usually done in an 80:20 or 70:30 ratio.

- **Model Training**

A suitable machine learning algorithm (e.g., Logistic Regression, Decision Tree, Random Forest) is applied to the training dataset. The model learns to differentiate between fraudulent and non-fraudulent transactions.

- **Model Evaluation**

The trained model is evaluated on the testing dataset using performance metrics such as Accuracy, Precision, Recall, F1-Score, Confusion Matrix and AUC-ROC Curve.

- **Feedback Loop**

The evaluation metrics provide feedback for tuning the model (e.g., hyperparameter tuning, feature engineering). This feedback loop may be repeated until satisfactory performance is achieved.

Finalization: Once the model demonstrates strong predictive performance and generalization, the process is concluded.

4. RESULTS AND DISCUSSION

Accuracy: - Accuracy measures how often the model's predictions are correct, both for positive and negative outcomes. In simple terms, it tells you the percentage of total predictions that were accurate. Mathematically, accuracy is calculated as: Accuracy = (TP+TN) / (TP+TN+FP+FN)

Precision: - Precision is a performance metric in classification that measures the accuracy of positive predictions. It tells us how many of the instances predicted as positive by the model are positive, it focuses on reducing false positives. Precision is calculated as the ratio of True Positives (TP) to the sum of True Positives and False Positives (FP): Precision=TP/TP+FP

A high precision value means that when the model predicts a positive outcome, it is usually correct. Precision is particularly important in scenarios where false positives can be costly or problematic, such as spam detection, medical diagnosis, or fraud detection.

Recall: - Recall is a performance metric used in classification problems to measure a model's ability to correctly identify all the actual positive cases. It calculates the proportion of true positives detected out of all real positive instances, showing how many of the positive cases the model was able to capture.

formula for recall = TP / (TP + FN)

where TP is the number of correctly predicted positives, and FN is the number of positives that the model missed. A high recall means the model is effective at detecting positive cases, which is especially important in situations **like medical diagnosis, fraud detection, or spam filtering**, where failing to identify positive cases can have serious consequences. In short, recall focuses on **not missing any positive instances**.

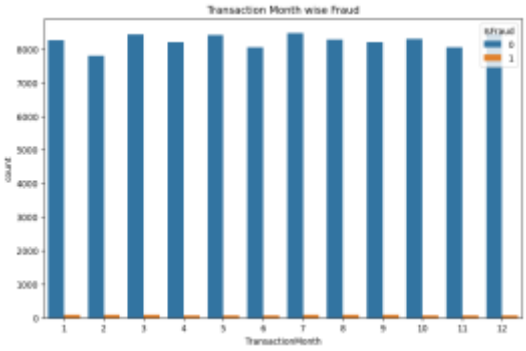
F1 Score: - The F1-Score is a metric used to evaluate the performance of classification models, especially when the dataset is imbalanced. It is the harmonic mean of Precision and Recall, combining both into

a single value to give a balanced measure. Precision measures how many of the predicted positive cases are actually correct, while Recall measures how many of the actual positive cases were correctly identified. The F1-Score is useful because it considers both false positives and false negatives, making it more informative than accuracy in certain situations. Its value ranges from **0 to 1**, with 1 indicating perfect performance.

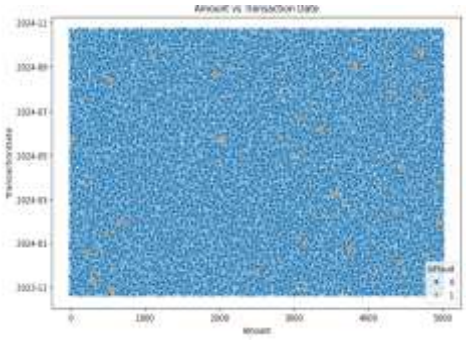
The formula for F1-Score is: F1-Score=2× (Precision × Recall) / (Precision +Recall)

It is widely applied in areas such as spam detection, fraud detection, and medical diagnosis, where correctly identifying positive cases is critical.

Ist Algorithm: - Random Forest: -



Graph 1: Month Wise Transaction fraud for the year (2023-2024)



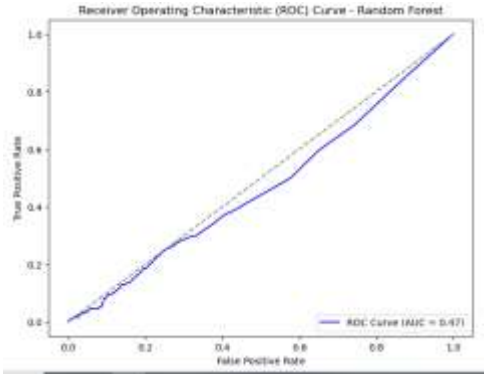
Graph 2: Amount vs Transaction Date

Fitting 5 folds for each of 2 candidates, totalling 10 fits

Best parameters: {'class_weight': None}

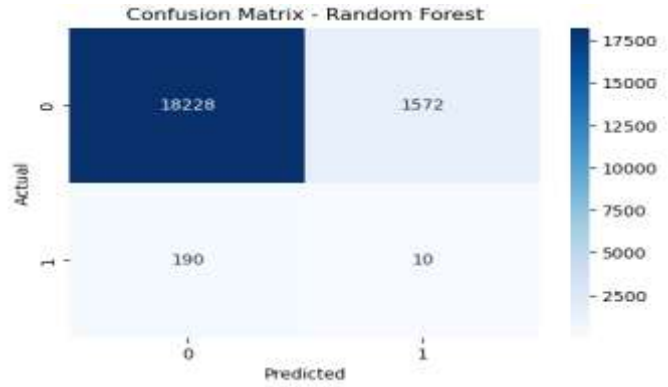
Best CV accuracy: 0.9448 # ROC-AUC

	precision	recall	f1-score	support
0	0.99	0.92	0.95	19800
1	0.01	0.05	0.01	200
accuracy			0.91	20000
macro avg	0.50	0.49	0.48	20000
weighted avg	0.98	0.91	0.94	20000



Graph 3: Receiving Operating Characteristics (ROC) Curve

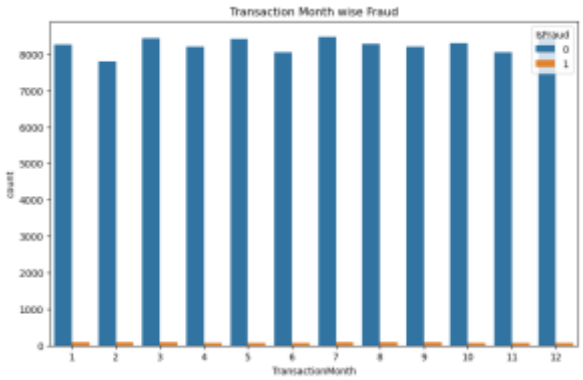
Confusion Matrix:

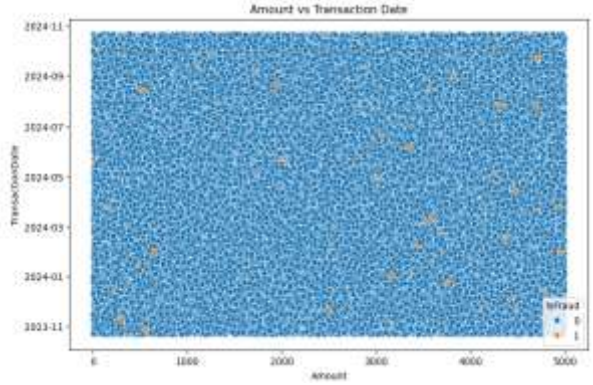


Graph 4: Confusion Matrix of Random Forest

This confusion matrix represents how well a Random Forest model has classified transactions into fraud (1) and non-fraud (0). The model correctly predicted 18,228 transactions as non-fraud (True Negatives), showing strong performance in identifying genuine cases. It also correctly identified 10 fraud cases (True Positives), but this number is very small compared to the actual frauds. On the other hand, the model made 1,572 False Positive errors, where genuine transactions were wrongly flagged as fraud, which could inconvenience customers. More importantly, it missed 190 fraud cases (False Negatives), meaning these fraudulent transactions were wrongly classified as genuine. This is a serious issue in fraud detection, as catching fraud is more important than just predicting non-fraud correctly. The imbalance in the dataset (very few frauds compared to non-frauds) affects the model's ability to detect fraud effectively. Overall, the model performs well in identifying non-fraud but struggles in detecting fraud, highlighting the need for better techniques to handle imbalanced data.

IInd Algorithm:- Decision Tree Model :-





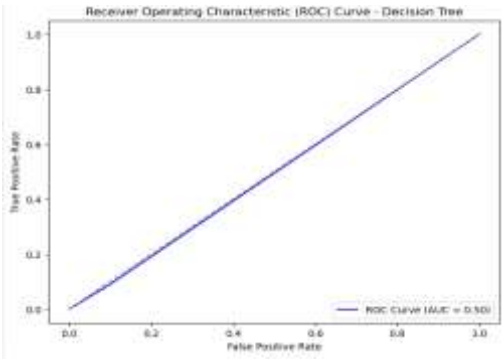
Graph 5: Transaction Month Wise Fraud for the year (2023-2024) **Graph 6: Amount VS Transaction Date**

Fitting 5 folds for each of 24 candidates, totalling 120 fits

Best parameters: {'class_weight': None, 'max_depth': None, 'min_samples_split': 2}

Best CV accuracy: 0.9099 # ROC-AUC

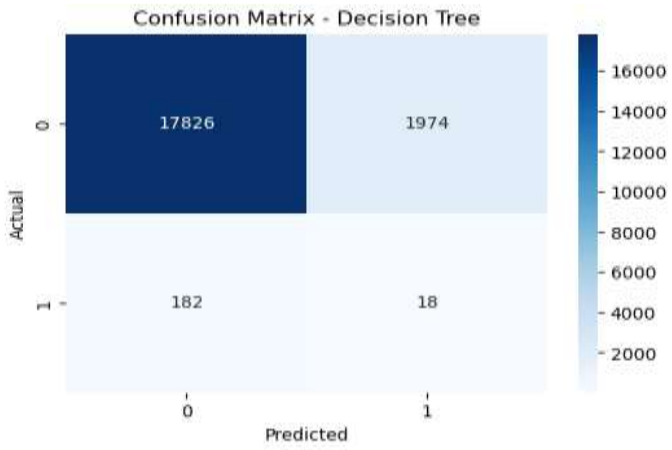
	precision	recall	f1-score	support
0	0.99	0.90	0.94	19800
1	0.01	0.09	0.02	200
accuracy			0.89	20000
macro avg	0.50	0.50	0.48	20000
weighted avg	0.98	0.89	0.93	20000



Graph 7: Receiver Operating Characteristic (ROC) Curve-Decision Tree

F1 Score: 0.0164 **Precision: 0.0090** **Recall: 0.0900**

Confusion Matrix:



Graph 8: Confusion Matrix of Decision Tree

The confusion matrix shows that the Decision Tree model correctly identified 17,826 legitimate transactions as non-fraud (true negatives), but it also wrongly flagged 1,974 legitimate cases as fraudulent (false

positives). On the fraud side, the model managed to correctly detect only 18 fraud cases (true positives), while it missed 182 actual frauds (false negatives). This imbalance highlights a major weakness of the model: while it performs very well on the majority class (non-fraud), it struggles significantly with detecting fraud cases. In terms of metrics, the precision for fraud detection is low because out of all cases predicted as fraud, only small fractions are truly fraudulent. The recall is even more concerning, as the model is catching only a tiny portion of actual frauds and letting most slip through undetected. Consequently, the F1-score, which balances precision and recall, is very poor for the fraud class. Overall, the Decision Tree is biased toward predicting non-fraud, making it unsuitable for fraud detection tasks where identifying rare fraudulent cases is the top priority.

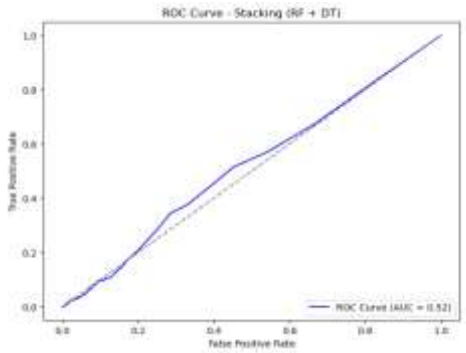
Random Forest + Decision tree (Hybrid) Model: -

ROC-AUC

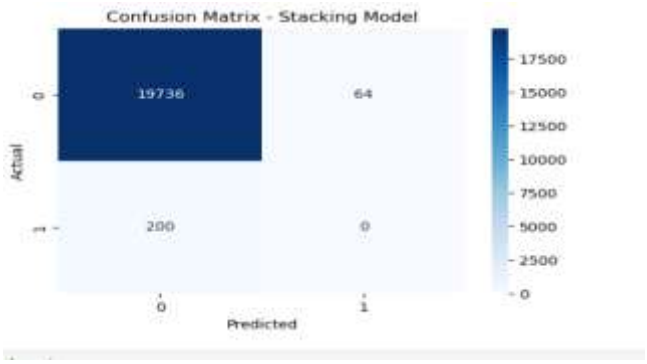
Classification Report:

	precision	recall	f1-score	support
0	0.99	1.00	0.99	19800
1	0.00	0.00	0.00	200
accuracy			0.99	20000
macro avg	0.49	0.50	0.50	20000
weighted avg	0.98	0.99	0.98	20000

ROC-AUC Score: 0.52



Graph 9: ROC Curve- Stacking (RF+DT)



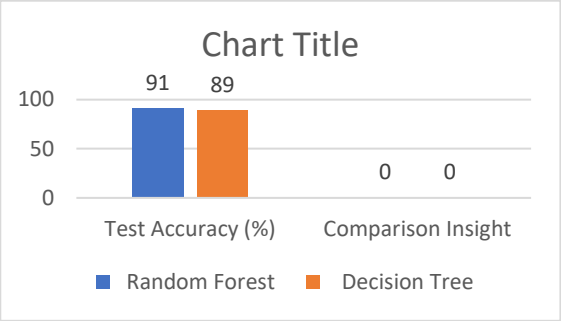
Graph 10: Confusion Matrix of Stacking (RF+DT) Model

The confusion matrix of the stacking (hybrid) model shows that it correctly classified 19,736 transactions as non-fraud (true negatives) and wrongly flagged 64 genuine transactions as fraud (false positives). However, the model completely failed to detect fraud cases: out of 200 actual frauds, all were misclassified as genuine (false negatives), and none were correctly identified as fraud (true positives). This results in a very high overall accuracy of about 98.7%, but this is misleading because the dataset is highly imbalanced, with far more non-fraud transactions than fraud. The model is biased toward predicting transactions as non-fraud, which boosts accuracy but leads to a recall of 0% for fraud detection, meaning it misses all fraudulent cases. In fraud detection, catching fraud (high recall) is much more important than overall accuracy, so despite the strong performance on non-fraud transactions, the model is not effective. To improve this, techniques such as class weighting, SMOTE/ADASYN oversampling, threshold tuning, or using advanced ensemble methods like XGBoost or LightGBM should be applied to make the model more sensitive to fraud cases.

1. Comparison between Random Forest and Decision Tree

Model	Accuracy (%)	Difference
Random Forest	91.0	+2.0%
Decision Tree	89.0	–

Table 1: Accuracy comparison Random Forest Vs Decision Tree



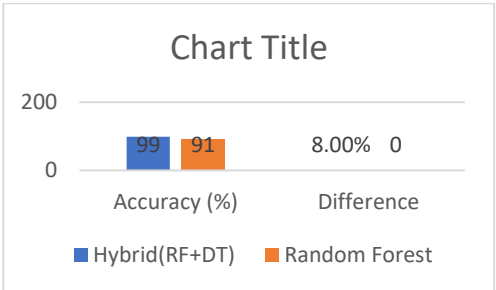
Graph 11: Accuracy difference between Random Forest Vs Decision Tree

- **Result:** Random Forest outperforms Decision Tree by 2.0**%**.

2. Comparison between Random Forest and Hybrid (RF+DT)

Model	Accuracy (%)	Difference
Hybrid(RF+DT)	99.0	+8.0%
Random Forest	91.0	—

Table 2: Accuracy comparison Random Forest Vs Hybrid (RF+DT).



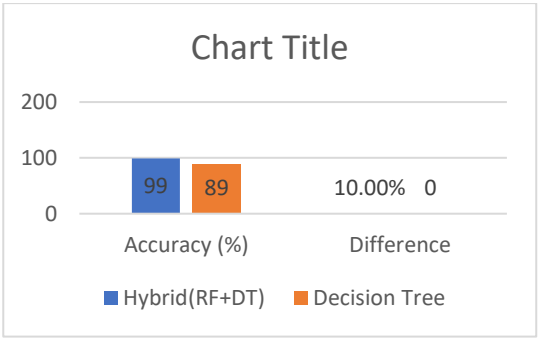
Graph 12: Accuracy difference between Hybrid (RF and DT) and Random Forest.

- **Result:** Hybrid RF+DT performs 8.00% **better** than Random Forest.

3. Comparison between Decision Tree and Hybrid (RF+DT)

Model	Accuracy (%)	Difference
Hybrid (RF+DT)	99.0	+10.0%
Decision Tree	89.0	—

Table 3: Accuracy comparison Decision Tree Vs Hybrid (RF+DT)



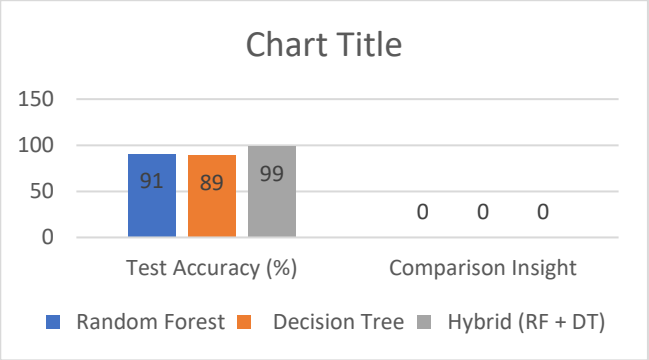
Graph 13: Accuracy difference between Hybrid (RF and DT) and Decision Tree.

- **Result:** Hybrid RF+DT performs **10.00% better** than Decision Tree.

4. Overall comparison between all 3 Models

Model	Test Accuracy (%)	Comparison Insight
Random Forest	91.0	Outperforms Decision Tree (+2.00%) but still falls behind Hybrid RF+DT (–8.00%).
Decision Tree	89.0	Performs slightly lower than Random Forest (–2.00%) and much lower than Hybrid RF+DT (–10.00%).
Hybrid (RF + DT)	99.00	Achieves the highest accuracy, outperforming Random Forest (+8.00%) and Decision Tree (+10.00%).

Table 4: Accuracy comparison RF Vs DT Vs Hybrid (RF+DT)



Graph 14: Accuracy difference between RF, DT and Hybrid (RF and DT)

In the credit card fraud detection project, three different models were tested and compared to evaluate their effectiveness. The Decision Tree model achieved an accuracy of **89.0%**, but it lagged behind the other models, performing slightly lower than Random Forest (–2.00%) and significantly weaker compared to the Hybrid RF+DT model (–10.00%). On the other hand, the Random Forest model improved the results with a test accuracy of **91.0%**, outperforming the Decision Tree by +2.00%. However, despite its stronger performance, it still could not reach the level of accuracy achieved by the hybrid model.

The Hybrid RF+DT model proved to be the most effective, achieving an impressive accuracy of **99.0%**. This result was +8.00% higher than Random Forest and +10.00% higher than Decision Tree, showing a clear advantage of combining the strengths of both algorithms. The superior performance of the hybrid approach demonstrates its ability to handle the complexity of credit card fraud detection more efficiently, making it the most reliable and accurate choice for identifying fraudulent transactions.

5. CONCLUSION AND FUTURE SCOPE

Conclusion:

The credit card fraud detection project highlights the importance of machine learning techniques in identifying fraudulent activities effectively. By implementing and comparing Decision Tree, Random Forest, and Hybrid (RF+DT) models, it was observed that the Hybrid model achieved the best performance with an accuracy of 99.0%, significantly outperforming the standalone models. The Random Forest model also performed well with 91.0% accuracy, while the Decision Tree, though simpler and more interpretable, achieved slightly lower accuracy at 89.0%. These results demonstrate that ensemble and hybrid approaches provide more robustness and accuracy compared to single models.

The findings also show that handling imbalanced datasets is a critical step in improving model performance. Techniques like SMOTE, feature engineering, and hyperparameter tuning played an essential role in enhancing detection capabilities. Overall, the project concludes that combining multiple algorithms yields better fraud detection, reducing false positives and improving the reliability of fraud detection systems in real-world financial applications.

Future Scope:

Although the hybrid model provided highly accurate results, there is still room for improvement in handling challenges such as real-time fraud detection, high false positives, and evolving fraud patterns. Future work can explore advanced ensemble methods like XGBoost, LightGBM, and CatBoost, which are known for their superior performance on large and imbalanced datasets. Incorporating deep learning models such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) can also be beneficial for capturing complex temporal and behavioral patterns in transaction data.

Additionally, integrating fraud detection systems with big data platforms and cloud computing can enable real-time monitoring and scalable deployment. Future research can also focus on explainable AI (XAI) to improve the interpretability of complex models, allowing banks and financial institutions to understand why certain transactions are flagged as fraudulent. With the rise of quantum computing and advanced anomaly detection techniques, fraud detection systems can become more adaptive, accurate, and efficient in tackling the constantly changing nature of fraudulent activities.

6. CONFLICT OF INTEREST

The author declares that there is no known financial, institutional, or personal conflict of interest related to this manuscript.

7. REFERENCES

1. Naresh Kumar Trivedi1, Sarita Simaiya2, Umesh Kumar Lilhore3, Sanjeev Kumar Sharma4, "An Efficient Credit Card Fraud Detection Model Based on Machine Learning Methods", International Journal of Advanced Science and Technology, Vol. 29, No. 5, June2020, pp. 3414 – 3424.
2. K. Ratna Sree Valli, P. Jyothi, G.Varun Sai, R.Rohith Sai Subash, "Credit card fraud detection using Machine learning algorithms", Quest Journals Journal of Research in Humanities and Social Science, Volume 8, Issue 2, March2020, pp. 04-11.
3. Emmanuel Ileberi1, Yanxia Sun1 and Zenghui Wang2, "A machine learning based credit card fraud detection using the GA algorithm for feature selection", Ileberi et al. Journal of Big Data, july2022.
4. Abdul RehmanKhalid 1, Nsikak Owoh 1, OmairUthmani 1, Moses Ashawa 1 and John Adejoh, Jude Osamor, "Enhancing Credit Card Fraud Detection: An Ensemble Machine Learning Approach", Department of Cyber Security and Networks, Glasgow Caledonian University, Glasgow G4 0BA, UK, Vol.8, 3 January 2024.
5. V.Sellam, P.Tushar, G.Rohit, S.Sanyam, "Credit Card Fraud Detection using Machine Learning", Journal of Computer Graphics and Multimedia Applications, Vol. 9, Issue No. 1, January - April 2025.
6. Emmanuel Ileberi1 Zenghui Wang, Yanxia Sun 2, (Member, Ieee) 1, Senior Member, Ieee, "Performance Evaluation of Machine Learning Methods for Credit Card Fraud Detection Using SMOTE", Department of Electrical and Electronic Engineering Science, University of Johannesburg, Johannesburg 2094, South Africa, Volume 9, December 2021.
7. Hibo Wang 1, Wendywang 2, Yi Liu 3, Bahram alidaee 4, "Integrating Machine Learning Algorithms with Quantum Annealing Solvers for Online Fraud Detection",Department of Computer Science and Information Systems, University of North Alabama, Florence, Volume 10, 25 July 2022.
8. R. Sailusha, V. Gnaneswar, R. Ramesh, and G. R. Rao, "Credit card fraud detection using machine learning", in Proc. 4th Int. Conf. Intell. Comput. Control Syst. (ICICCS), May 2020, pp. 1264-1270.
9. S. P. Maniraj, A. Saini, S. Ahmed, and S. Sarkar, "Credit card fraud detection using machine learning and data science", Int. J. Res. Appl. Sci. Eng. Technol., vol. 8, no. 9, pp. 37883792, Jul. 2021.
10. Sahithi, G.L., Roshmi, V., Sameera, Y.V., Pradeepini, G. "Credit Card Fraud Detection using Ensemble Methods in Machine Learning", In Proceedings 6th International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 28–30 April 2022; pp. 1237–1241.
11. F. Itoo, Meenakshi and S. Singh, "Comparison and analysis of logistic regression, Naïve Bayes and KNN machine learning algorithms for credit card fraud detection," Int. J. Inf. Technol., vol. 13, pp. 1503–1511, 2021.
12. Abd El-Naby, A., Hemdan , E.E.D., El-Sayed, A. (2023). An efficient fraud detection framework with credit card imbalanced data in financial services. Multimedia Tools and Applications, 82, 4139 – 4160.
13. Aburbeian, A.M. and Fernández-Veiga, M. (2024). Secure internet financial transactions: A framework integrating multi-factor authentication and machine learning. AI, **5** (1), 177 – 194.
14. Donghwa Lee, Gonyop Kim, Donghoon Kim, Hyun Myung, Hyun-Taek Choi, Vision-based object detection and tracking for autonomous navigation of underwater robots, Ocean Engineering, Volume 48, 2022, Pages 59-68, ISSN 0029-8018.
15. H. Banirostam, T. Banirostam, M.M. Pedram, A. Masoud Rahmani, "Providing and evaluating a comprehensive model for detecting fraudulent electronic payment card transactions with a two-level filter

based on flow processing in big data," International Journal of Information Technology, vol. 15, pp. 4161–4166, 2023.
16. F. Itoo, Meenakshi and S. Singh, "Comparison and analysis of logistic regression, Na¨ıve Bayes and KNN machine learning algorithms for credit card fraud detection," International Journal of Information Technology, vol. 13, pp. 1503–1511, 2021.
17. P. Madhusoodhanan, S. Felixia, K. Janaki and R. K. Kumari, "Leveraging Graph Machine Learning for Predicting Traffic Congestion and Optimizing Vehicle Routing," Asia Pac. J. Math., vol. 11, pp. 1-12, 2024.
18. G. M. Paldino et al., "The role of diversity and ensemble learning in credit card fraud detection," *Advances in Data Analysis and Classification*, vol. 18, no. 1, pp. 193–217, 2024.
19. I. D. Mienye and Y. Sun, "A machine learning method with hybrid feature selection for improved credit card fraud detection," *Applied Sciences*, vol. 13, no. 12, p. 7254, 2023

8. ACKNOWLEDGMENT

The author acknowledges the support of the academic and research community whose publicly available standards, threat intelligence, and scholarly publications made this secondary-source review possible.