FMDB *Transactions on Sustainable Technoprise Letters*

Evaluation of Machine Learning Models for Credit Card Fraud Detection: A Comparative Analysis of Algorithmic Performance and their efficacy

Sonjoy Ranjon Das[1,*], Antigoni Kruti[2], Rajan Devkota[3], Rejwan Bin Sulaiman[4]

[1,2,4]Department of Computer Science and Technology, Northumbria University, England, United Kingdom.
[3]Department of Computer Engineering, Tribhuvan University, Kirtipur, Nepal.
sonjoy.das@northumbria.ac.uk[1], antigoni.kruti@northumbria.ac.uk[2], pas075bct033@wrc.edu.npuk[3],
rejwan.sulaiman@northumbria.ac.uk[4]

Abstract: Credit card fraud has increased vulnerability effects due to the large usage functions for customers due to innovative technologies and communication patterns. This article presents a review and important analysis of credit card fraud detection and prediction of fraudulent transactions based on cutting-edge research. The study provides a limited investigation into deep machine learning to address the effects of data issues on credit card fraud detection through the design of robust solutions. This study aims to develop a mechanism with classifiers, such as Artificial Neural Network (ANN), Support Vector Machine (SVM), and Naive Byes, that contain vectors of information sequence properties, structure and mechanisms. Simultaneously, diverse experiments are developed to analyze the proposed approach to datasets. The framework enhances a comprehensive financial security diverse approach to suspicious financial activities on stakeholders' assets. It emphasizes the significance of mitigation and detection capabilities for potential threats to safeguard financial transactions. The outcome of this research demonstrates a robust solution for real scenarios of credit card fraud detection, considering model abilities with high accuracy rates that address the limitations of integrated factors.

Keywords: Credit Card Fraud Detection; Machine Learning; Artificial Neutral Network; Support Vector Machine; Naïve Byes; Class Imbalance; Exploratory Data Analysis; Logistic Regression and Random Forest.

Received on: 19/10/2022, **Revised on:** 21/01/2023, **Accepted on:** 05/04/2023, **Published on:** 07/12/2023

Cite as: S. Ranjon Das, A. Kruti, R. Devkota, and R. Bin Sulaiman, "Evaluation of Machine Learning Models for Credit Card Fraud Detection: A Comparative Analysis of Algorithmic Performance and their efficacy," *FMDB Transactions on Sustainable Technoprise Letters*, vol. 1, no. 2, pp. 70–81, 2023.

1. Introduction

Credit cards are widely used in e-commerce and online transactions, leading to the expansion of multiple credit card fraud forms and mechanisms. Fraudsters are applying sophisticated models to conduct illegal transactions due to data fishing and fake cards produced to mimic the legitimate behaviour of cardholders, which results in financial losses. Eventually, it is crucial to be up-to-date and to have robust credit card fraud detection systems by classifying the transaction as legitimate and illegitimate. This paper contributes to the reported solutions analysis of malicious transactions. It emphasises developing innovative technologies involving machine learning algorithms and recently advanced integrations such as real-time detection aspects, class imbalance, and big data technologies. In-depth, this research paper shows the state-of-the-art solutions, including machine learning implementation and testing. This paper is organized into six sections as follows. The first section is the introduction that illustrates this academic work's motivation and aims. The second section presents the methodology and the research involving the scheme illustrated in Figure 1 to divide data on training and testing to provide deep analysis methods via comparative analysis. The third section displays the scope and the background of this research. The fourth section presents

[*]Corresponding author.

the proposed work focused on the deep learning solutions of credit card detection systems countermeasures. The fifth section emphasizes the fraud detection dataset analysis provided with deep testing metrics. The sixth section concludes the paper with open issues and highlights the results of the work.

2. Methodology

Figure 1 illustrates the methodology presented in this paper, outlining the credit card fraud detection process. The methodology encompasses several steps, including data collection and selecting an appropriate model for the project dataset.

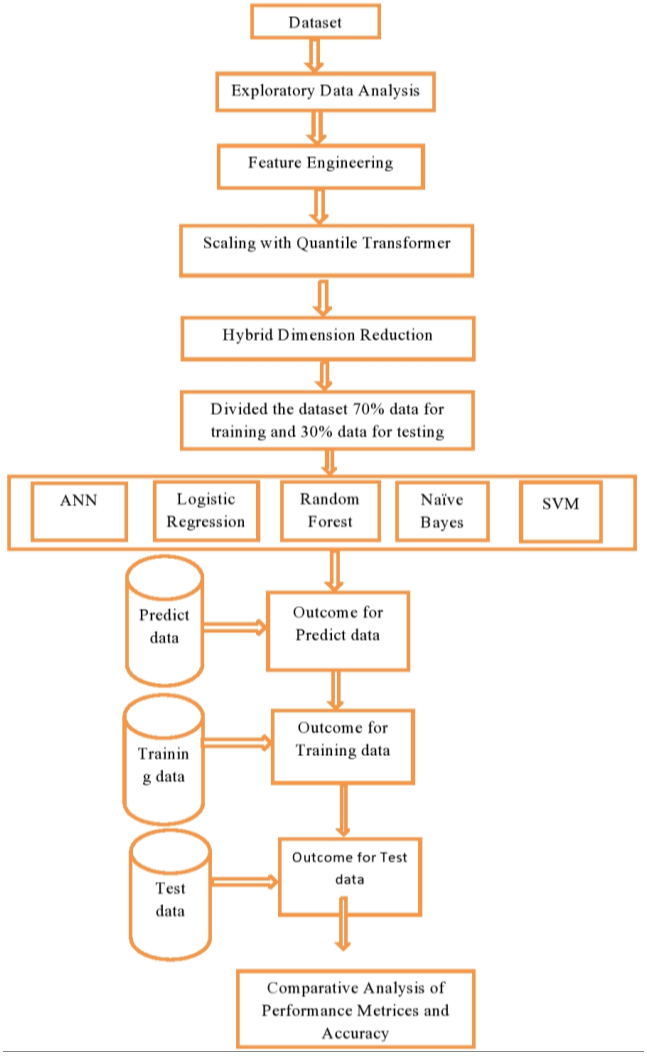

Figure 1: Methodology of Credit Card Fraud Detection

3. Result

The credit card transactions carried out by European cardholders during September 2013 are stored in this dataset. Within two days, 284,807 transactions were recorded, among which 492 were classified as frauds. This renders the dataset heavily imbalanced as the frauds constitute only 0.172% of all the transactions [1]. The dataset consists exclusively of numerical input variables, which are the outcome of a PCA transformation. The original features and additional background information about the dataset cannot be disclosed due to confidentiality restrictions. The features V1, V2, and V28 are the principal components acquired through PCA. Only two features, 'Time' and 'Amount,' have not been transformed by PCA. 'Time' denotes the duration between each transaction and the first transaction recorded in the dataset, whereas 'Amount' represents the transaction amount. This feature can be utilized for instance-dependent cost-sensitive learning. The response variable is 'Class,' and it is assigned a value of 1 for fraud and 0 for other transactions.

Exploratory Data Analysis (EDA): In this project, we conducted Exploratory Data Analysis (EDA) to gain insights into the dataset, identify patterns, and detect potential issues or anomalies. First, we loaded the dataset and utilized functions like head (), describe (), and shape to obtain a data summary. During this step, we observed no missing or null values in the dataset. However, we quickly recognized that the dataset was highly imbalanced, with one class significantly outnumbering the other. This class imbalance could impact our analysis and model performance, so addressing this issue appropriately during the data pre-processing and modelling phases is essential.

Feature engineering: Feature engineering plays a pivotal role in credit card fraud detection as it aims to enhance the fraud detection model's performance [2]. In this project, we have focused on several important feature engineering techniques, including:

Removing Duplicate Data: Ensuring data cleanliness and accuracy is vital. As part of this process, we have identified and eliminated duplicate records in the dataset. Removing duplicates is essential to prevent biased training and evaluation of the model.

Separating Dependent and Independent Features: Properly distinguishing between dependent (target) and independent (predictor) features is crucial. The dependent feature represents the fraud label, indicating whether a transaction is fraudulent. The independent features are used to analyze patterns and make predictions related to fraud.

Scaling with Quantile Transformer: In this project, we have employed the Quantile Transformer, a non-linear transformation technique, to scale the features of our dataset. This process involves mapping the data to a uniform or Gaussian distribution, depending on the desired output distribution [3]. Using the Quantile Transformer, we can assess the assumptions made by linear methods like Min-Max scaling and Standardization.

By applying the Quantile Transformer, we aim to bring the data closer to a uniform or Gaussian distribution, which could be advantageous for specific machine learning algorithms. We can improve the algorithms' performance by transforming the data and facilitating better model training and predictions.

Hybrid dimensionality reduction: In this project, we have adopted a Hybrid dimensionality reduction approach, which combines multiple dimensionality reduction techniques to extract significant features and patterns from the dataset [4]. The main objective is to enhance the fraud detection model's performance by leveraging the strengths of different methods.

For dimensionality reduction, we employed an autoencoder technique followed by PCA. This combined approach allows us to create a more informative and compact representation of the data, which can lead to improved credit card fraud detection performance [5]. As a non-linear technique, the autoencoder can capture intricate relationships in the data, while PCA, as a linear method, further refines the reduced representation.

By applying this hybrid approach, we can effectively distil essential information from the original dataset, making it more manageable and easier for the fraud detection model to discern fraudulent patterns from legitimate transactions. The resulting model is expected to achieve better accuracy and detection capabilities, contributing to a more robust credit card fraud detection system.

Dividing the dataset: Following the sampling of the dataset, this paper allocated 70% of the data for training purposes and the remaining 30% for testing. During the training phase, this research employed various supervised machine learning algorithms such as Logistic Regression, Random Forest, SVM, Naïve Bayes, XGBoost and Artificial Neural Networks. Here is a brief description of all algorithms:

Logistic Regression: This research chose this algorithm because it is specifically designed for classification problems, the type of problem presented in this dataset. As a result, this paper opted for it over the linear regression algorithm. It is effortless to implement in both binary and multivariate classification problems [6].

Our dataset is a binary classification problem with a single predictor variable. This study wants to predict whether a transaction is fraudulent or not fraudulent based on the class column where 0 means non-fraudulent and 1 means fraudulent transaction of the predictor variable. This study represents the predictor variable as x and the binary outcome as y (0 or 1).

In logistic regression, we assume that the relationship between the predictor variable and the outcome can be modelled using the logistic function (also known as the sigmoid function) [7]. The logistic function is defined as:

$$f(x) = 1 / (1 + e^{(-x)})$$

This function maps any real-valued values of input features and then determines the most discriminative feature to split the data at each tree node.

Random Forest: The algorithm commences by randomly sampling instances from a provided dataset. It creates a decision tree for each sampled instance and generates a prediction result from each tree, as illustrated in Figure 2. Subsequently, the algorithm performs a voting process on the predicted results, selecting the outcome with the highest number of votes as the final prediction result [8].

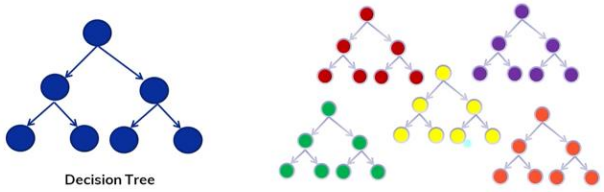

Figure 2: Random Forest creation based on Decision Tree

Figure 3 presents the working flowchart of the Random Forest algorithm. The process begins with providing a training set divided into different subsets. Each subset undergoes classification, assigned to classes 1, 2, and 3. Ultimately, based on the results obtained through bootstrapping, the Random Forest algorithm produces the final output.

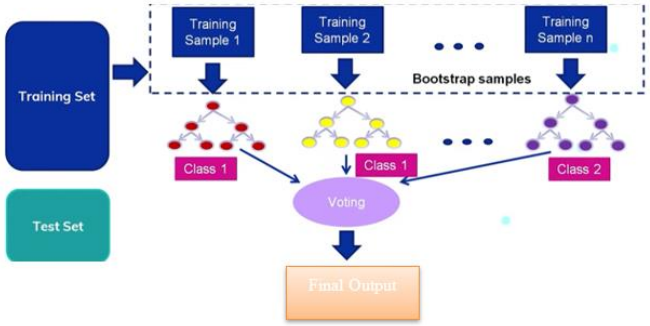

Figure 3: Workflow of Random Forest

Naive Bayes: The implementation of the Bayes Theorem was chosen for this analysis due to its simplicity, speed, and efficiency, requiring less training data. Additionally, it exhibits scalability and is capable of handling large datasets. The algorithm operates by calculating the probability of an event happening based on the probability of another event that has already occurred [9].

$$P(A/B) = (P(B/A)\,P(A)) / P(B)$$

Where, P (A) – Priority of A P (B) – Priority of B

P (A/B) – Posteriori priority of B

SVM: Support Vector Machine (SVM) is a powerful supervised machine learning algorithm that can be used for both classification and regression problems. It works by finding a hyperplane that optimally separates the data into distinct classes, using a method known as the maximum margin. SVM is known for its effectiveness in handling high-dimensional data and can also handle non-linearly separable data using kernel functions [8]. However, SVM can be computationally expensive and requires careful selection of hyperparameters. SVM is a versatile algorithm that can be applied to regression and classification problems. In our specific case, where the problem is focused on classification (fraudulent or non-fraudulent transactions), we have chosen to utilize SVM on this dataset.

Figure 4 illustrates the working mechanism of Support Vector Machines (SVM). The process begins with a labelled dataset used for model training. Subsequently, the trained model is utilized to make predictions. Finally, the output, representing the predicted results, is presented:

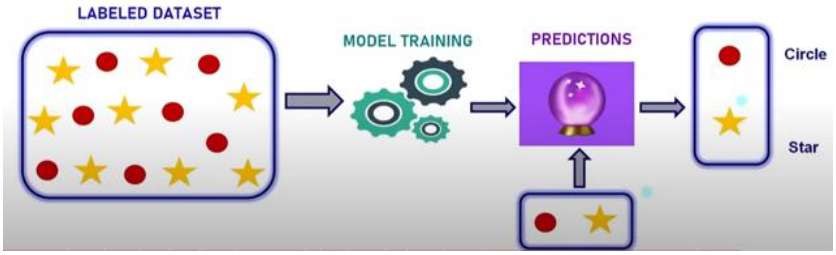

Figure 4: Working Mechanism of SVM

ANN: Artificial Neural Network (ANN) is a flexible machine learning algorithm capable of being applied to both supervised and unsupervised learning tasks. Its versatility enables it to handle various tasks, including classification, regression, and clustering. This adaptability and broad applicability make ANN a popular choice for numerous real-world applications, which further justifies its selection for implementation in this paper [9]. ANN has been selected as the implementation choice in this paper to enhance the accuracy of various models on the dataset. In addition, ANN is particularly known for its ability to learn complex patterns and relationships in data and can handle large and high-dimensional datasets [10]. However, ANN can be computationally expensive and requires careful selection of architecture and hyperparameters to achieve optimal performance.

AdaBoost, short for Adaptive Boosting, is a machine learning algorithm primarily designed for binary classification tasks [1]. In AdaBoost, every instance in the training dataset is assigned an initial weight. The initial weight for each training instance, denoted as weight (xi), is set to 1/n, where xi represents the i-th training instance, and n is the total number of training instances in the dataset.

Finding the best algorithm involves systematically evaluating and comparing different algorithms based on their performance on a given task or problem. Here are the general steps to find the best algorithm [6]:

- Define the Problem
- Prepare the Data
- Select a Set of Algorithms
- Split the Data
- Train and Evaluate Models
- Tune Hyperparameters
- Select the Best Algorithm
- Test the Best Algorithm

Iterate if Needed:

- Predict the data: After training and testing our model, it is essential to evaluate its accuracy in predicting fraudulent and normal transactions [7].
- Outcome of the Predicted Data: In the results section, this project will provide Figure 5, showing the performance of each algorithm [12].
- Training Data: Once we have trained our model on 70% of the data, we will use the remaining 30% to evaluate its performance metrics and accuracy [11].
- Outcome of the Training Data: In the results section, we will provide a table showing the model's performance on the training data.
- Testing the data: We will proceed with the testing phase once the model has been trained.
- Outcome of Testing: The performance metrics and results for each algorithm will be visualized in graphs.

Accuracy Result: After evaluating the performance of each algorithm, the results are presented in terms of accuracy. Accuracy represents the proportion of correctly classified instances out of the total instances in the dataset. The best algorithm, which exhibits the highest accuracy among all the evaluated algorithms, is identified by comparing the accuracy scores of different algorithms. Identifying the best algorithm based on accuracy allows for selecting the model that achieves the highest level of correct predictions, indicating its effectiveness in solving the given problem [9].

Evaluation: Different algorithms are evaluated using various measures to assess different aspects. These measures serve as criteria for evaluating various proposed methods. Credit card fraud detection researchers commonly employ False Positive

(FP), False Negative (FN), True Positive (TP), and True Negative (TN) and their relationships as quantities to compare the accuracy of different approaches [10]. The definitions of these parameters are provided below.

- True Positive (TP): The true positive rate signifies the proportion of correctly classified fraudulent transactions out of all the actual fraudulent transactions. It is calculated as TP divided by the sum of TP and FN.
- True Negative (TN): The true negative rate represents the proportion of accurately classified normal transactions out of all the actual normal transactions. It is calculated as TN divided by the sum of TN and FP.
- False Positive (FP): The false positive rate indicates the proportion of wrongly classified non-fraudulent transactions as fraudulent. It is calculated as FP divided by the sum of FP and TN.
- False Negative (FN): The false negative rate indicates the proportion of wrongly classified non-fraudulent transactions as normal. It is calculated as FN divided by the sum of FN and TP.

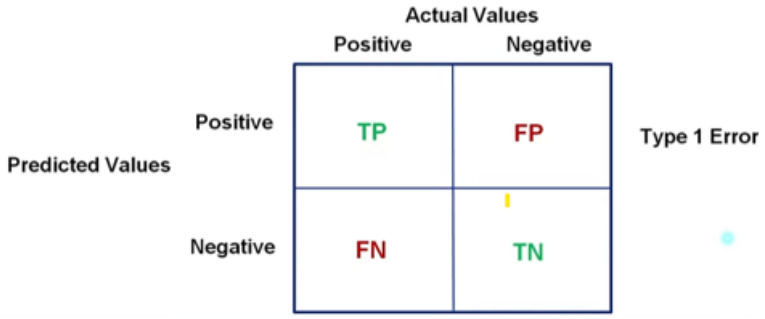

Figure 5: Evaluation Metrics for Credit Card Fraud

Confusion Matrix: The confusion matrix offers valuable insights into the performance of a predictive model by revealing how each class is predicted accurately or inaccurately and the types of errors made. When dealing with a two-class classification problem, typically consisting of negative and positive classes, the confusion matrix provides a clear and well-defined structure [11]. Each cell in the matrix represents a category with a well-established name, allowing for a comprehensive analysis of the model's predictive capabilities (Figure 6).

	Predicted	
	Fraudulent	Non-Fraudulent
Actual		
Fraudulent	TP	FN
Non-Fraudulent	FP	TN

Figure 6: Metrics explanation

Accuracy: Accuracy is a commonly used performance metric in classification tasks, representing the percentage of instances correctly classified by a model [9].

$$Accuracy = (TP + TN) / (TP + TN + FP + FN)$$

Precision: Precision measures the proportion of correctly predicted positive instances (true positives) out of all instances predicted as positive (true positives + false positives) [9]. It focuses on the accuracy of positive predictions and is calculated as follows:

$$Precision = TP / (TP + FP)$$

A high precision indicates a low false positive rate, meaning that the model has a low tendency to incorrectly label negative instances as positive.

Recall (Sensitivity or True Positive Rate): Recall measures the proportion of correctly predicted positive instances (true positives) out of all actual positive instances (true positives + false negatives). It focuses on the model's ability to correctly identify positive instances and is calculated as follows [13]:

$$Recall = TP / (TP + FN)$$

A high recall indicates a low false negative rate, meaning the model can effectively identify positive instances without missing many.

F1 Score: The F1 score is a harmonic mean of precision and recall, providing a balanced evaluation metric that considers both precision and recall. It combines precision and recall into a single value and is calculated as follows [10]:

$$F1\ Score = 2 * (Precision * Recall) / (Precision + Recall)$$

The F1 score ranges from 0 to 1, where 1 represents perfect precision and recall, and 0 indicates poor performance in either precision or recall.

Support: Support refers to the number of instances in the dataset that belong to a specific class. It represents the actual occurrences of that class in the dataset. Support is a valuable metric that helps identify any imbalances in the training data [13]. If there is a significant difference in the number of instances between different classes, it may highlight potential weaknesses in the classifier's performance. Addressing imbalanced support could involve stratified sampling or rebalancing the dataset. It's important to note that the support value remains consistent across different models and serves as a diagnostic tool to assess the evaluation process rather than being influenced by the models themselves.

4. Credit Card Fraud Detection System Taxonomy and Challenges

The Anatomy of Credit Card Fraud Detection System Design: Credit card fraud detection systems (FDS) develop successful procedures and processes to detect and reject suspicious transactions, as illustrated in Figure 7. The critical investigations of a comprehensive credit card fraud detection framework from Andrea et al. suggest a layer control mechanism for industrial partners' operational activities such as the terminal, investigators, scoring rules, the module of blocking rules and the Data-Driven model (DDM) [14].

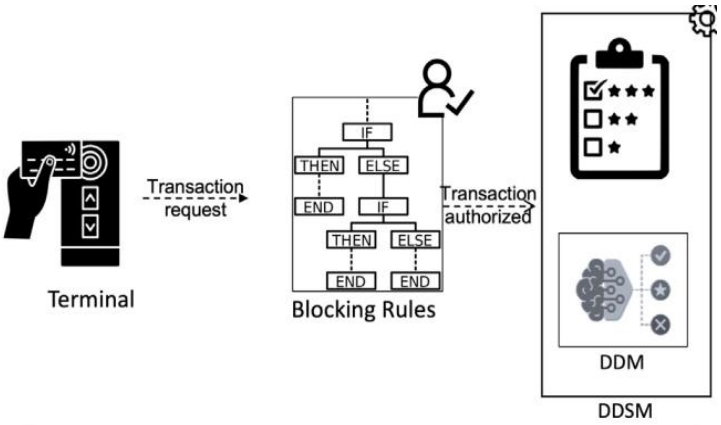

Figure 7: Credit Card Fraud Detection System Design

The terminal allows the customers to execute their transactions. Then, those transactions are delivered in real-time to the blocking module for primary protection checks based on logical rules. This step is fundamental to controlling if-then-else confidential rules of fraud pattern detection that human investigators could discover. Following this, the DDM consists of historical transaction processing to develop a classifier or statistical model to identify fraudulent credit card transactions. Estimating the legal transactions through the scores is crucial because in cases where the score overextends the well-estimated threshold, the transaction is rejected and evaluated by experts on further analysis and examinations [7]. The DDM should be qualified to be fully automated from a labelled dataset. Then, if the cardholder reports any fraudulent transaction that the classifier model does not identify, the case should be investigated by human experts. Investigators are involved in all the human intervention in double-checking transaction activities.

On the other hand, Jain [13] modified the previous model to establish a new, improved automated scoring model in the same category as DDM as a combined data-driven scoring model (DDSM). The new design element establishes a real-time process that emphasizes and evaluates FDS processes. On the contrary, Karthik et al. [14] designed a near real-time DDM and scoring

processing. The FDS practices are useful for quantifying the degree of fraud detection. FDS operates with fully automated distribution operations because the number of investigators is limited, and suspicious investigating practice is time-consuming.

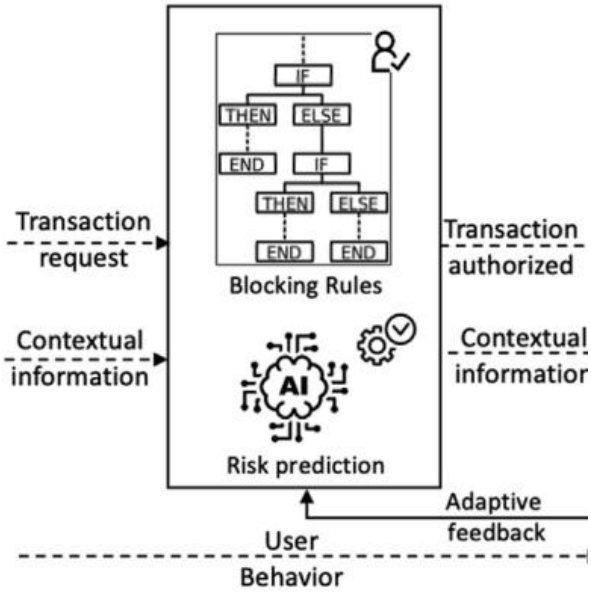

Figure 8: Credit Card Automated Design

Figure 8 shows the new FDS system architecture, which offers complementing blocking rules predicated on the intelligent model to estimate transaction risks [9]. The technology only operates when the rules are met, reducing the danger of suspicious activity. It rejects early if blocking rules are confirmed and the operation is hazardous. After risk mitigation, the transaction is transmitted to 3DSM, a distributed DDSM. The 3DSM distributes the learning module using cutting-edge technology. The 3DMS minimises human intervention and prioritises numerous data inputs through learning module updates that include contextual data and user behaviour like location and typing [15].

Fig. 9 shows the key credit card fraud detection challenges, which include data-related, security, implementation, and deployment. FDS uses the DDSM module to create successful detection models with high-quality datasets [17]. In several areas, researchers found class imbalance or skewed datasets. Most machine learning algorithms use data categorization for an equal sample quantity per class, making imbalanced data categories difficult to forecast. Due to data imbalance, predictive algorithms may perform poorly on minorities.

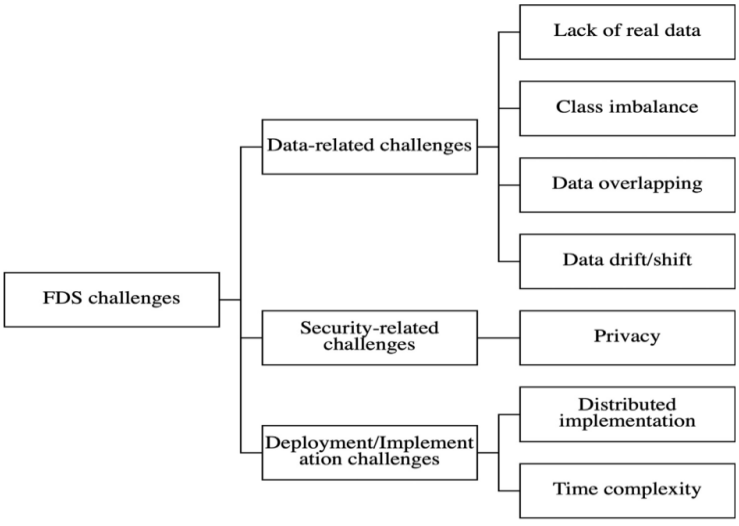

Figure 9: Credit Card Fraud Detection Issues

According to Zhang and Huang [16], the traditional machine-learning algorithm requires a balanced dataset to maintain its function for the minor class. Specifically, this approach concludes that the performance of most machine learning algorithms is degraded from the unbalanced dataset, for instance, Support Vector Machine (SVM) [11]. The technique proposed by Li et al. [15] to solve this problem consists of adapting algorithms to compile unbalanced datasets or developing pre-processing sampling methods. On the other hand, the lack of real data is another challenge of the unavailability of sufficient labelled data. Sometimes, the data demands supplementary efforts to identify and describe all the data rows. Indeed, one common method to detect fraud is anomaly detection. Consequently, it depends on the variation of user behaviour and the user's historical actions. Samples from different classes could appear on the equivalent data space areas, where information is hard to distinguish. In the cases where fraudulent or non-fraudulent transactions of credit card fraud detection overlap, the classifier performance has independent effects, affecting a linear deterioration in the classifier performance. Li et al. [15] report three main steps to solve this problem. It begins with the original dataset divided into a non-overlapping and overlapping subset. Then, the under-sampling method is utilised to remove the samples of the overlapping subsets from the majority class. In the final step, the classifier detects and maintains the minority samples.

Data drift is another inevitable problem with detecting credit fraud on machine learning accuracy. This challenge involves the input data from changes that cause the deterioration via the degradation of machine learning performance. Data drift could be initiated by covariance or data relationship modifications. This particularly occurs during credit card fraud detection because of consumer behaviour changes. According to Asha and Kr [3], all those modifications caused variations to the distribution of data on training and testing data, and the efforts to optimize the performance and the system accuracy aim to minimize the risks due to those modifications. Indeed, developing a credit card fraud detection model is essential to requiring and selecting proper features from raw data by extracting financial transaction aggregation from cardholder patterns.

Problems with security The credit card transaction system's detection models should protect privacy. It should keep attackers from understanding the learning model or cardholder patterns. Panthakkan et al. [17] suggested a federated learning system based on cardholder behaviours to address security problems. Banks can train fraud detection models during data delivery over local storage with this model. Combining computed local updates over local detection models creates the detection system. All banks could benefit from the integrated model without sharing their data. Preventing fraudsters from stealing users' identities is crucial. Plakandaras et al. [18] addressed cardholder identity fraud detection in cyber-physical systems (CPS) with restricted customers. A dependable server can proxy and delegate client authentication. A detection protocol for authentication servers was proposed to outsource trusted authentication processes.

5. Implementation or deployment challenges

The strong FDS should make scaling robust systems easy. Monolithic banking design has various drawbacks, including the high expense of scaling up to add consumers. Banks can switch to cloud deployment because cloud computing technology has matured elastic scalability. A good FDS system considers latency, heterogeneous data integration, and new frameworks and technologies to meet the infrastructure of new-generation banking systems. Online transactions are real-time, therefore time complexity necessitates the FDS to make millisecond decisions. The system may have a long delay in detecting fraud from excluded platform users. Femila Roseline et al. [19] say FDS should make massive data streaming decisions instantly. The FDS should detect fraud before approving the transaction. However, Sailusha et al. [21] argue that information fetching delays transactions, making real-time detection difficult. It may affect usability and consumer satisfaction. New streaming data technologies like Apache Sparks that deliver real-time analytics could provide larger transaction responsiveness for fraud detection [20].

6. Machine Learning Techniques of Credit Card FDS Detections and Countermeasures

Transactions made with credit cards can have the potential to constitute an unbalanced dataset that includes fraudulent transactions. There is a concern regarding data imbalance, which can be created by challenges in data storage or by an intrinsic property that occurs when the dataset has an unequal class distribution due to issues including privacy or increased costs.

6.1. Oversampling techniques

Carcillo et al. [7] present a generic technique to generate new input data with K-mean clustering from the minor class of credit card theft. This will improve system accuracy and stabilise the dataset. Autoencoders are recommended for selecting discriminative dataset features for fraudulent instances. Dornadula and Geetha, 2019 used Synthetic Minority Oversampling (SMOTE) and One-class SVM to create cardholder profiles for unbalanced data. They can measure system performance using Mathews' correlation coefficient (MCC) applications. Rtayli and Enneya supported this technique in 2020 with hybrid system solutions that use SMOTE to describe and solve the class imbalance problem. Furthermore, Sailusha et al. [21] decided to increase the machine learning models' effectivity for credit card fraud detection by further establishing this approach with two

joint resampling techniques such as SMOTE and adaptive Synthetic (ADASYN), which is an improvement of SMOTE applied to evaluate and compare the dataset performance via for machine learning algorithms such as logR, BT, RF and KNN. On the other hand, Carcillo et al. [7] decided to oversample their dataset with varying sizes of samples, reaching 20,000 to reach a high level of performance by using sklearn.util.resample utility that is focused on bootstrapping.

6.2. Under sampling techniques

Tiwari et al. [22] found that random under-sampling might tackle class scalability and imbalance issues better than deep learning models. Different datasets yielded 10:1 non-fraudulent ratios. However, Li et al. [15] offered a divide-and-conquer technique to class imbalance with overlap. They create an anomaly detection model for minority and many majority class samples. They formed an overlapping subset with a low imbalance ratio. Since subsets overlap, they utilise a non-linear classifier to classify samples. They proposed using Dynamic Weighted Entropy (DWE) to measure quality, taking into account the ratio class imbalance of overlapping groups and the number of excluded minorities.

6.3. Hybrid techniques

It has been suggested by Trivedi et al. [23] as the results of testing many classifiers for massive data vis Random Oversampling (ROS) and SMOTE, SVM-SMOTE, SMOTEENN and SMOTETomek types examinations with Random Under sampling (RUS). Also, BBE, which is internal balance with RUS with SMOTE, was tested to develop a hybrid balancing model. The results outlined that the hybrid techniques are scalable and optimize the best system performance. Furthermore, Udeze et al. [24] support this approach by using SMOTE to oversample fraudulent instances, RUS and condensed nearest neighbour (CNN) for genuine records under-sampling. Moreover, Wasserbacher and Spindler [25] applied different machine learning algorithms in a comparative experiment study to tackle the imbalance dataset classification problem to detect and mitigate credit card fraud with methods of over and under-sampling. The studies have obtained experimental insights into using ML techniques to detect card fraud with imbalanced data. The authors used NearMiss as an under-sampling technique and SMOTE as a sampling technique.

6.4. Databases and Testing Parameters for Fraud

In our study, we trained different models to analyse the recall values for detecting false predictions. We employed traditional machine learning algorithms and a Multi-Layer Perceptron (ANN). Our primary focus was detecting False Negatives, specifically aiming for a model that accurately predicts fraud while minimizing confusion with non-fraudulent cases. Therefore, the key parameters for model selection were the Recall for the Fraud class and the Macro Average. The results of the analysis are illustrated in the table 1 below.

Table 1: The results of the analysis

Modal	Recall Score [Class Based]	Recall [Macro Avg.]
ANN	0.81	0.81
Logistic Regression	0.76	0.76
Random Forest	0.11	0.56
Xgboost Classifier	0.36	0.68
Support Vector Classifier	0.76	0.81
Naïve Bayes	0.09	0.54

Recall Score [Class Based]: This metric indicates the recall score for the fraud class. Based on this metric, the models that performed well are ANN, Support Vector Classifier, and Logistic Regression. ANN achieved the highest recall score of 0.81, while Logistic Regression and SVC tied for the second-highest recall score of 0.76.

Recall Score [Macro Avg]: This metric represents the average recall score across fraud and non-fraud classes, giving equal weight to each class. According to this metric, the top-performing models are ANN, Support Vector Classifier, and Logistic Regression, all with a recall score of 0.81.

In conclusion, from our results, considering both the recall score Class Based and Macro Average, ANN, Logistic Regression, and Support Vector Classifiers demonstrate strong performance. These models exhibit good performance in terms of recall score for both class-based recall and macro average [26]. The usage of precision, sensitivity error rate and currency rate have measured the performance of the suggested system. As a result of that analysis, a model was created to optimize current credit card fraud detection tactics with accuracy based on the Multi-Layer Perception [27]. The outcome with the boosting strategy outperformed the ANN, SVM and Naïve Bayes methods via the comparison for better outcome. The limitations include

validating the sampling technique for extremely messy data without ordinary distribution [8]. The study could not guarantee similar performance for different models with various data types.

7. Conclusion

In conclusion, the proposed research highlights the significance of Credit Card Fraud Detection mechanisms via machine learning techniques to provide an effective solution for dynamic tasks on financial transactions. The results of security framework performance have been achieved via integrating algorithms such as Artificial Neural Networks, Support Vector Machine, Radom Forest, and Naïve Bayes to classify financial transactions and ensure a robust recognition system. The SVM and Radom Forest algorithms are crucial to identify and emerge any potential threat to complex models with linear relationships on complex datasets. On the other hand, the Random Forest Algorithm enable the compound outcome tree aggregation to mitigate unnoticed data risks. The Naïve Bayes probabilistic model ensures a security solution based on the independence assumption approach. The combination of Random Forest for discovering mechanism, Naïve Nayes for the straightforwardness of independent features, ANN for sample evaluation to distinguish indirect indicators and SVM for non-linear allocations on the legitimate margin of dataset develop an adaptable, efficient and accurate framework for credit card fraud detection system. The research contributes to developing comprehensive and integrated showcase evaluation for diverse conditions of data collection evaluation of the model outcome factors. The financial landscape is continuously evolving. This research is capable of providing a deep understanding of machine learning techniques on multifaced credit card financial issues and ensuring defence mechanisms and security methods for customers.

Acknowledgement: I extend my profound gratitude to Dr Faizan Ahmad, my research project supervisor, and the resources provided by Northumbria University, which played a pivotal role in completing this research endeavour. I would also like to sincerely thank my wife and my four-year-old daughter for their unwavering support during the challenging phases of this research. The dedication to this project is deeply rooted in the memory of my younger sister, who tragically succumbed to lung cancer on December 1, 2023.

Data Availability Statement: The datasets produced and scrutinized during this study can be accessed at https://www.kaggle.com/. Embracing principles of transparency and open science, this project invites fellow researchers to utilize, reproduce, and extend the project's data for further exploration. For inquiries regarding access to the data, please contact Sonjoy Ranjon Das via email at sonjoytheanlyst@gmail.com.

Funding Statement: This research was self-funded, as the author independently pursued this project out of personal interest during the research project proposal course at the university.

Conflicts of Interest Statement: The authors declare that no conflicts of interest exist concerning the publication of this paper. This project affirms that the research was conducted with integrity, free from any external influence that might compromise the study's objectivity or the reporting of its results. Additionally, all references from which the data have been collected are duly provided in this project.

Ethics and Consent Statement: This study adhered to the ethical standards and guidelines Northumbria University, London, set forth. All research protocols involving human subjects were reviewed and approved by the university's ethics committee and supervisor. Informed consent was obtained from all participants involved in the study, and they were assured of the confidentiality and anonymity of their information. The ethical conduct of this research aligns with the principles outlined in the research involving human subjects.

References

1. A. El-Naby, A. Hemdan, and E. E. D. El-Sayed, “An efficient fraud detection framework with credit card imbalanced data in financial services,” Multimedia Tools and Applications, vol. 82, no. 3, pp. 4139–4160, 2023.
2. M. H. Ahmed and A. H. Butt, A Review: Credit Card Fraud Detection in Banks using Machine Learning Algorithms. 2023, Press.
3. R. B. Asha and S. K. Kr, “Credit card fraud detection using artificial neural network,” Global Transitions Proceedings, vol. 2, no. 1, pp. 35–41, 2021.
4. S. Bagga, A. Goyal, N. Gupta, and A. Goyal, “Credit card fraud detection using pipeling and ensemble learning,” Procedia Computer Science, vol. 173, pp. 104–112, 2020.
5. A. Bhanusri, K. R. S. Valli, P. Jyothi, G. V. Sai, and R. Rohith, “Credit card fraud detection using Machine learning algorithms,” Journal of Research in Humanities and Social Science, vol. 8, no. 2, pp. 4–11, 2020.

6. R. Bin Sulaiman, V. Schetinin, and P. Sant, “Review of machine learning approach on credit card fraud detection,” Human-Centric Intelligent Systems, vol. 2, no. 1–2, pp. 55–68, 2022.
7. F. Carcillo, Y. A. Le Borgne, O. Caelen, Y. Kessaci, F. Oblé, and G. Bontempi, “Combining unsupervised and supervised learning in credit card fraud detection,” Information sciences, vol. 557, pp. 317–331, 2021.
8. 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, no. 4, pp. 1503–1511, 2021.
9. V. N. Dornadula and S. Geetha, “Credit card fraud detection using machine learning algorithms,” Procedia computer science, vol. 165, pp. 631–641, 2019.
10. P. Gupta, A. Varshney, M. R. Khan, R. Ahmed, M. Shuaib, and S. Alam, “Unbalanced Credit Card Fraud Detection Data: A Machine Learning-Oriented Comparative Study of Balancing Techniques,” Procedia Computer Science, vol. 218, pp. 2575–2584, 2023.
11. M. Habibpour et al., “Uncertainty-aware credit card fraud detection using deep learning,” Eng. Appl. Artif. Intell., vol. 123, no. 106248, p. 106248, 2023.
12. A. Izotova and A. Valiullin, “Comparison of Poisson process and machine learning algorithms approach for credit card fraud detection,” Procedia Computer Science, vol. 186, pp. 721–726, 2021.
13. S. Jain, A comparative analysis of various credit card fraud detection techniques Intelligent Decision Support System View project Saksham Kaksha View project. 2019.
14. V. S. S. Karthik, A. Mishra, and U. S. Reddy, “Credit card fraud detection by modelling behaviour pattern using hybrid ensemble model,” Arab. J. Sci. Eng., vol. 47, no. 2, pp. 1987–1997, 2022.
15. Z. Li, M. Huang, G. Liu, and C. Jiang, “A hybrid method with dynamic weighted entropy for handling the problem of class imbalance with overlap in credit card fraud detection,” Expert Syst. Appl., vol. 175, no. 114750, p. 114750, 2021.
16. Z. Zhang and S. Huang, “Credit card fraud detection via deep learning method using data balance tools,” in 2020 International Conference on Computer Science and Management Technology (ICCSMT), Shanghai, China, IEEE, 2020.
17. A. Panthakkan, N. Valappil, M. Appathil, S. Verma, W. Mansoor, and H. Al-Ahmad, “Performance Comparison of Credit Card Fraud Detection System using Machine Learning,” in 2022 5th International Conference on Signal Processing and Information Security (ICSPIS), Dubai, United Arab Emirates, IEEE, pp. 17–21, 2022.
18. V. Plakandaras, P. Gogas, T. Papadimitriou, and I. Tsamardinos, “Credit card fraud detection with automated machine learning systems,” Appl. Artif. Intell., vol. 36, no. 1, 2022.
19. J. Femila Roseline, G. Naidu, V. Samuthira Pandi, S. Alamelu alias Rajasree, and D. N. Mageswari, “Autonomous credit card fraud detection using machine learning approach,” Comput. Electr. Eng., vol. 102, no. 108132, p. 108132, 2022.
20. “ULB machine learning group,” Ulb.ac.be. [Online]. Available: http://mlg.ulb.ac.be. [Accessed: 10-Nov-2023].
21. R. Sailusha, V. Gnaneswar, R. Ramesh, and G. R. Rao, “Credit card fraud detection using machine learning,” in 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, IEEE, 2020.
22. P. Tiwari, S. Mehta, N. Sakhuja, J. Kumar, and A. K. Singh, “Credit card fraud detection using machine learning: A study,” arXiv [cs.AI], 2021, Press.
23. N. K. Trivedi, S. Simaiya, U. K. Lilhore, and S. K. Sharma, “An efficient credit card fraud detection model based on machine learning methods,” International Journal of Advanced Science and Technology, vol. 29, no. 5, pp. 3414–3424, 2020.
24. C. L. Udeze, I. E. Eteng, and A. E. Ibor, “Application of machine learning and resampling techniques to credit card fraud detection,” J. Nig. Soc. Phys. Sci., p. 769, 2022.
25. H. Wasserbacher and M. Spindler, “Machine learning for financial forecasting, planning and analysis: recent developments and pitfalls,” Digit. Finance, vol. 4, no. 1, pp. 63–88, 2022.
26. R. Arora, N. Dixit, and G. Dubey, “A review on fraud detection of credit cards through machine learning algorithms,” Jusst.org. [Online]. Available: https://jusst.org/wp-content/uploads/2023/01/A-Review-on-Fraud-Detection-of-Credit-Cards-through-Machine-Learning-Algorithms.pdf. [Accessed: 8-Nov-2023].
27. A. Cherif, A. Badhib, H. Ammar, S. Alshehri, M. Kalkatawi, and A. Imine, “Credit card fraud detection in the era of disruptive technologies: A systematic review,” J. King Saud Univ. - Comput. Inf. Sci., vol. 35, no. 1, pp. 145–174, 2023.