ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 11, Issue 9, September 2024

Credit Card Fraud Detection Using Machine Learning Algorithms

[1] Ruth C. Amanze, [2] Ernest E. Onuiri, [3] Oluwatobi O. Faromoju, [4] Airat F. Sorinolu

[1] [2] [3] [4] Department of Computer Science, School of Computing, Babcock University, Nigeria

Abstract*— This study aims to address the rising issue of credit card fraud by developing a machine learning model capable of identifying and preventing fraudulent transactions. The model works by analyzing transaction data to detect potential fraud, subsequently canceling the transaction and alerting the credit card owner. Credit card fraud detection is a classification problem, where various machine learning algorithms are applied to distinguish between legitimate and fraudulent transactions. The analysis emphasizes the importance of robust countermeasures due to the increasing use of credit cards globally. However, real-world implementation of such systems may face challenges, particularly in securing the cooperation of banks and addressing resource constraints. The study also highlights key dataset features that correlate with fraudulent behavior, with ensemble methods standing out as top-performing algorithms in terms of accuracy and efficiency.

Index Terms*— Algorithms, Credit Card Fraud, Classification, Data Analysis, Fraud Detection, Machine Learning, Predictive Modeling, Transaction Security.

I. INTRODUCTION

Credit card fraud is a form of identity theft that involves the unauthorized taking of someone else's credit card information to charge purchases to the account or remove funds from it [1]. The use of credit and debit cards, respectively, as a means of payment is very popular among online shoppers, who have a wide range of payment methods to choose from [2]. During a March 2017 survey of global online shoppers, 42 percent of respondents stated that they preferred to pay via credit card [3]. Suffice it to say, credit cards were the most popular payment method, ahead of electronic options such as PayPal [4]. In 2018, $24.26 billion was lost due to payment card fraud worldwide [5], with the United States leading as the most fraud-prone country [6], accounting for 38.6% of reported card fraud losses in 2018 [5].

There has been a proportional increase in the rate of credit card fraud and financial losses [7]. Credit card fraud affects consumers, merchants, and issuing banks [8]. Anyone who collects payments or customer information online is at risk of being targeted by thieves [9]. This is a significant issue that requires the attention of professional communities such as machine learning and data science, where automated solutions to this problem can be developed. Therefore, it is crucial to establish reliable techniques for detecting credit card fraud to proactively combat illegal activities involving credit cards.

II. RELATED WORKS

Ghosh and Reilly. [10] proposed a technique using a neural network to detect credit card fraud using data from a credit issuer. The neural network-based fraud detection system uses a dataset of labelled credit card account transactions. The authors dealt with many different forms of credit card fraud. Results show that their method detected more fraudulent accounts with fewer false positives. The system focuses on detection accuracy and speed.

John Akhilomen. [11] presented a data mining program for detecting cyber credit card fraud. Data mining has gained popularity in combatting credit card fraud due to its powerful artificial intelligence (AI) techniques and algorithms that can be deployed to identify or anticipate fraud through knowledge discovery from odd patterns produced from acquired data. A system model for detecting cyber credit card fraud is described and created in this work. This system uses a data mining supervised anomaly detection algorithm to identify fraud in a real-time internet transaction, rating the transaction as genuine, suspicious fraud, or illegitimate. A neural network is used implement the anomaly detection algorithm.

Chan et al. [12] proposed a system of leveraging distributed data mining to identify credit card fraud. They claimed that large-scale data-mining approaches can advance business practice. Scalable strategies for analyzing enormous volumes of transaction data and effectively computing fraud detectors in real time is a critical issue, particularly in e-commerce. Aside from scalability and efficiency, the fraud-detection job has technical issues such as skewed training data distributions and non-uniform cost per error, both of which have received little attention in the knowledge discovery and data mining communities. The article's writers examine and assess many strategies that handle these three major difficulties at the same time. The authors examine various strategies that handle these three major difficulties at the same time. Their suggested approaches for merging numerous learnt fraud detectors under a "cost model" are broad and beneficial; findings also show that the authors were able to drastically minimize fraud loss using distributed data

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 11, Issue 9, September 2024

mining of fraud models.

Abhinav Srivastava et al. [13] proposed utilizing the hidden Markov model to identify credit card fraud (HMM). They employed a hidden Markov model (HMM) to represent the processes in credit card transaction processing and explain how it may be used to detect fraud. An HMM is initially trained using a cardholder's regular behavior. If an incoming credit card transaction is not accepted with a sufficiently high probability by the trained HMM, it is considered fraudulent. At the same time, we make every effort to guarantee that legitimate transactions are not refused.

Sam Maes et al. [14] proposed automated credit card fraud detection employing two machine learning algorithms. Artificial neural networks and Bayesian belief networks were used to solve to the problem. The results show a significant improvement on real-world financial data and future directions.

Various facts about detection systems, data mining, and machine learning have been analyzed in the paper, now, the logic behind describing credit card fraud as a major problem has become abundantly clear. Year after year as the use of credit cards has increased exponentially, fraudsters have devised more creative methods of defrauding people of their money. This has motivated the digital security industry to make increasingly sophisticated research on methods to combat such activities. Artificial intelligence technology has helped in developing revolutionary models that have made finding anomalistic patterns in transactional databases much easier [15]. The models are even being used by major financial institutions that are looking for reliable methods of protecting their customer's interests. This is the justification why such technologies need to be studied and more efficient problem-solving methods need to be curated and developed.

III. MATERIALS AND METHODS

This paper presents an approach of using different machine learning algorithms one at a time to detect irregular activities during a transaction.

Firstly, we obtained our dataset from Kaggle, a repository of datasets in various fields. The dataset contained 15 total features including the dependent variable for the project, the "Is Fraud" column. The features were year, month, day, time, amount, use chip, merchant name, merchant city, merchant state, zip, Merchant Category Codes (MCC), and errors. These features were all equipped with either a 1 or 0 value for the dependent variable.

We performed necessary measures needed to clean our dataset by eliminating unnecessary or irrelevant observations as that helped to speed up the analysis and make the dataset more manageable and performant.

We performed univariate and bivariate analysis which was a mix of histograms, density plots and boxplots to describe the columns of the dataset and to see the effect one column has on another column. The following were the visualizations for the analysis on the dataset:

Fig. 1. Visualizations of the Analysis of the Dataset

This graph shows the times at which transactions took place.

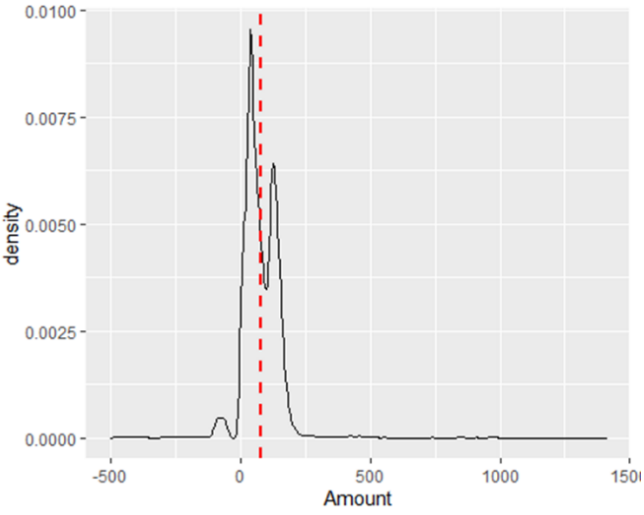

Fig. 2. Graph showing transaction times

This graph represents a density plot that shows the amount that was transacted.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 11, Issue 9, September 2024

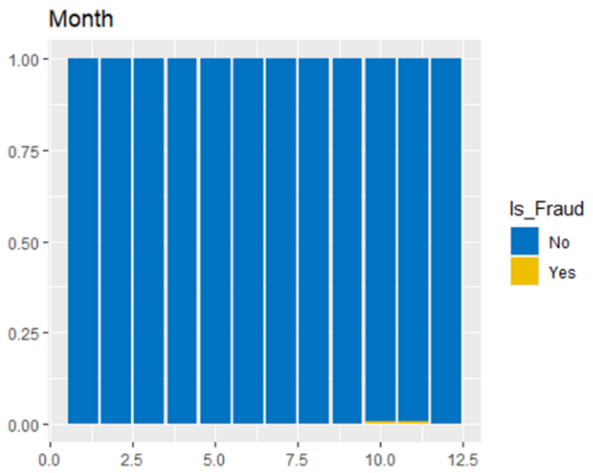

Fig. 3. Density plot showing amounts transacted

This graph shows a stacked bar chart for Month against the dependent variable (Is_Fraud) and it can be seen that most fraudulent transactions occurred in the 10th and 11th months (October and November).

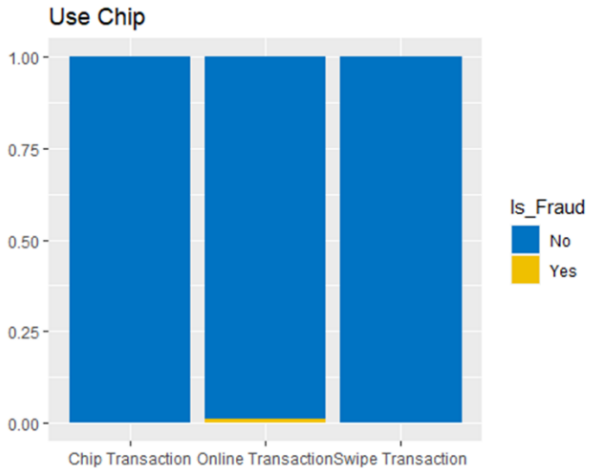

Fig. 4. Graph showing fraudulent activities occurred by month

This graph shows the number of transactions that took place either with a chip, online or swipe transaction. It shows that online transaction had the most fraudulent transactions.

After this analysis, we plotted a correlation matrix for the dataset which is formed based on Pearson's Correlation Coefficient to show how related each variable is to each other, especially the dependent variable. Also, the correlation tests were calculated and tested at a significance level of 0.05. The correlation matrix is shown below:

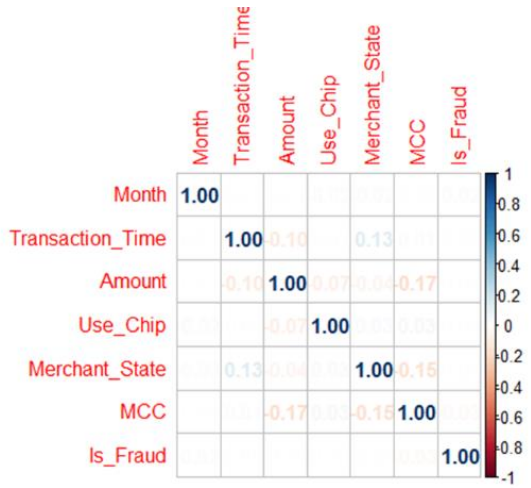

Fig. 5. Correlation Matrix of the dataset based on Pears'n's Correlation Coefficient.

This correlation matrix shows there was little correlation between the columns and the dependent variable. In fact, the results of the correlation matrix were incredibly low.

```
# Transaction features
cor.test(creditcard$Is_Fraud,creditcard$Month)$p.value #significant
cor.test(creditcard$Is_Fraud,creditcard$Transaction_Time)$p.value #Not significant
cor.test(creditcard$Is_Fraud,creditcard$Amount)$p.value #Not significant
cor.test(creditcard$Is_Fraud,creditcard$Use_Chip)$p.value #Not significant
cor.test(creditcard$Is_Fraud,creditcard$Merchant_State)$p.value #Not significant
cor.test(creditcard$Is_Fraud,creditcard$MCC)$p.value #significant
```

```
[1] 0.02958589
[1] 0.6401455
[1] 0.7037189
[1] 0.5151786
[1] 0.3782305
[1] 0.0001286413
```

Fig. 6. Results of the correlation matrix

There existed correlation significance between label the 3 out of 5 features in the dataset at p = 0.05. The features transaction time and amount had correlation significance above the 0.05 cut off.

The dataset was then formatted and processed, so steps were taken to build the ideal model based on the given dataset for a credit card fraud detection system.

Credit card fraud datasets are typically heavily imbalanced and that is the case in this dataset. Imbalanced datasets are very problematic for classification problems because the selected machine learning model will be biased towards the majority class. In the case of the selected dataset, the machine learning models were likely to predict that most transactions are non-fraudulent transactions without applying any machine learning logic. The accuracy score for such a model may return very good results but the Area Under the Curve (AUC) for the Receiver Operating Characteristic (ROC) curve tells an alternative story. ROC curve is a measure for telling how good a machine learning model performs by plotting its True Positive Rate (TPR) against its False Positive Rate (FPR).

To solve the class imbalance problem, a combination of random oversampling and random under-sampling techniques were employed. The Random Over-Sampling

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 11, Issue 9, September 2024

Examples (ROSE) package was used to balance the two dependent variables in the dataset.

```
creditcard_balanced_over <- ovun.sample(Is_Fraud ~ ., data = creditcard,
method = "both",N = 19963)$data
table(creditcard_balanced_over$Is_Fraud)

##
##     0     1
## 10079  9884
```

Fig. 7. Number of entries in each dependent variable after oversampling and undersampling

Furthermore, we performed feature selection as feature selection is an important aspect of any machine learning project. It involved using an algorithm to decide what features of the dataset were useful for predicting the outcome of the dependent variable. In this project, Boruta, a feature selection package in R was used. Boruta works by it introducing randomness into the data set by producing shuffled copies of all the features of interest (these are referred to as shadow features).

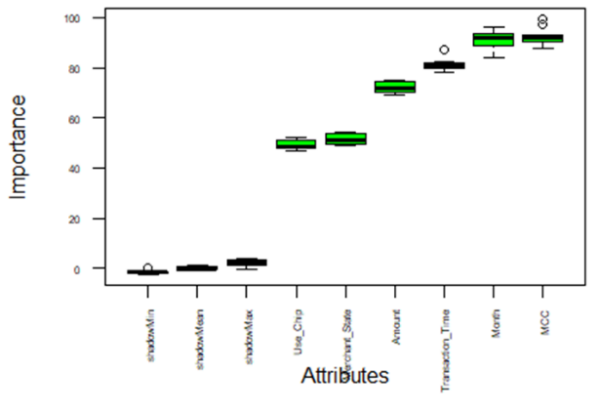

Fig. 8. Feature Selection done using Boruta

Boruta decided that all features were important for predicting the dependent variable for the dataset. The Month and MCC variables were the most important variables according to the Boruta algorithm.

The dataset was then split into the train and test datasets to prepare for the machine learning section. 70% was used for the training dataset and 30% was used for the testing dataset. The train dataset was named dataset, while the test dataset was named validation.

```
prop.table(table(creditcard_balanced_over$Is_Fraud))

##
##         0         1
## 0.504884 0.495116

prop.table(table(dataset$Is_Fraud))

##
##          0          1
## 0.5049016 0.4950984

prop.table(table(validation$Is_Fraud))

##
##         0         1
## 0.504843 0.495157
```

Fig. 9. Distribution of fraud labels into different datasets.

The datasets were checked to see if the proportionality between the datasets were evenly split. This were to ensure that the dependent variables were not widely dissimilar to each other as it would've caused inconsistent results during model building.

To build the machine learning model, 8 machine learning algorithms were evaluated.

The linear algorithms used were:

- Logistic Regression (LG)
- Linear Discriminant Analysis (LDA)
- Naive Bayes (NB)

The non-linear algorithms used were:

- K-Nearest Neighbors (KNN)
- Classification and Regression Trees (CART)
- Support Vector Machine (SVM)

The ensemble methods used were:

- Random Forest (RF)
- XGBoost (XGB)

Analysis was done using 10-fold cross validation with 3 repeats.

```
## Accuracy
##           Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## LG   0.7831067 0.8014668 0.8086552 0.8089205 0.8192884 0.8303508    0
## LDA  0.7730852 0.7932761 0.7992128 0.8012159 0.8105878 0.8210451    0
## NB   0.9549034 0.9649499 0.9677881 0.9676086 0.9717199 0.9763948    0
## SVM  0.7738010 0.7925167 0.7967799 0.7973050 0.8041492 0.8217609    0
## XGB  0.9992842 1.0000000 1.0000000 0.9998569 1.0000000 1.0000000    0
## KNN  0.9928418 0.9957082 0.9967801 0.9968275 0.9978537 1.0000000    0
## CART 0.8496779 0.8716023 0.9027182 0.8936657 0.9146539 0.9306152    0
## RF   0.9985694 0.9992842 1.0000000 0.9996184 1.0000000 1.0000000    0
##
## Kappa
##           Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## LG   0.5664934 0.6031297 0.6174503 0.6180329 0.6387282 0.6608303    0
## LDA  0.5465479 0.5868127 0.5986702 0.6026896 0.6213707 0.6422880    0
## NB   0.9098746 0.9299420 0.9356113 0.9352531 0.9434650 0.9528073    0
## SVM  0.5479724 0.5854132 0.5938448 0.5949049 0.6084805 0.6437141    0
## XGB  0.9985683 1.0000000 1.0000000 0.9997138 1.0000000 1.0000000    0
## KNN  0.9856842 0.9914162 0.9935600 0.9936549 0.9957072 1.0000000    0
## CART 0.6995299 0.7433958 0.8057958 0.7875587 0.8295683 0.8614090    0
## RF   0.9971386 0.9985683 1.0000000 0.9992367 1.0000000 1.0000000    0
```

Fig. 10. Metrics after building the model

The top three performing algorithms were RF, XGB and KNN with mean accuracies of 0.9996, 0.9995 and 0.9969 respectively. We then proceeded by tuning the three algorithms to help improve the accuracy. The tuning for the random forest machine learning model is shown below:

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 11, Issue 9, September 2024

mtry	Accuracy	Kappa
1	0.9920332	0.9840676
2	0.9992367	0.9984733
3	0.9997853	0.9995706
4	0.9996422	0.9992844
5	0.9996184	0.9992367
6	0.9995707	0.9991413
7	0.9995707	0.9991413
8	0.9995707	0.9991413
9	0.9995707	0.9991413
10	0.9995707	0.9991413

Fig. 11. Tuning of the Random Forest Model

It showed that accuracy improved the accuracy of the RF model.

The tuning for the XGBoost machine learning model is shown below:

Accuracy	Kappa
0.9998092	0.9996184

Fig. 12. Tuning of the XGBoost model

It showed that tuning did not improve the accuracy of the XGB model.

Tuning for the KNN machine learning model is shown below:

k	Accuracy	Kappa
1	0.9991173	0.9982345
2	0.9983302	0.9966602
3	0.9979485	0.9958968
4	0.9973045	0.9946088
5	0.9968275	0.9936549
6	0.9961359	0.9922717
7	0.9956112	0.9912224
8	0.9950627	0.9901255
9	0.9946095	0.9892193
10	0.9941086	0.9882176

Fig. 13. Tuning of the KNN model

It showed that tuning improved the accuracy of the KNN model.

Random forest produced the highest results for the preliminary test, so the model was selected for the testing in the project. The confusion matrix for the model was constructed:

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 3021    0
##          1    2 2965
##
##                Accuracy : 0.9997
##                  95% CI : (0.9988, 1)
##     No Information Rate : 0.5048
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.9993
##
##  Mcnemar's Test P-Value : 0.4795
##
##             Sensitivity : 0.9993
##             Specificity : 1.0000
##          Pos Pred Value : 1.0000
##          Neg Pred Value : 0.9993
##              Prevalence : 0.5048
##          Detection Rate : 0.5045
##    Detection Prevalence : 0.5045
##       Balanced Accuracy : 0.9997
##
##        'Positive' Class : 0
```

Fig. 14. The Model's Confusion Matrix

RF model performed at an accuracy of 0.9997 with only 2 error calls leading to a specificity of 1.000 but a sensitivity score of 0.9993. Kappa was very high at 0.9993.

Now, we calculated the Area Under Curve (AUC) of the Receiver Operating Characteristic (ROC) to evaluate the overall accuracy of the model.

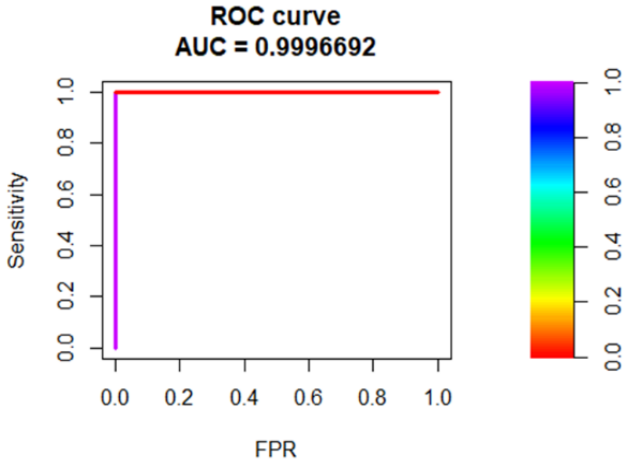

Fig. 15. Visualization of the ROC curve

The area under the curve retuned a result of 0.9996692, meaning the model performed very well.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 11, Issue 9, September 2024

IV. DISCUSSION

The results from implementation show that a coherent system has been built for detecting fraudulent transactions using machine learning. For this to be accomplished, data cleaning, data exploration, further data analysis and model building were the required steps taken. By every relevant measure for a machine learning project, the model has been able to perform to an excellent level, culminating in a ROC score of 0.999. This is an excellent result for a machine learning model in any field, especially in the field of credit card fraud detection.

The original hypothesis for the paper assumed that the address verification system feature had no effect on the likelihood of the transaction being fraudulent. The findings from this research show that the address verification system did not have a significant effect on the state of the Is_Fraud variable. This was evident from the feature selection section where Boruta deemed it not significantly suitable for the machine learning process.

V. CONCLUSION

Fraud detection is a complex problem that necessitates more thought and practice when using machine learning algorithms. This ensures that the customer's money is secure and can no longer be easily tampered with. Credit card usage is increasing in every aspect of daily life. Credit card fraud will also rise. An effective and efficient credit card detection system has been designed to improve the security of transaction systems. The system is built on the back of a machine learning model that predicts whether a transaction is fraudulent or not based on a few relevant variables.

REFERENCES

[1] F. K. Alarfaj, I. Malik, H. U. Khan, N. Almusallam, M. Ramzan, and M. Ahmed, "Credit Card Fraud Detection Using State-of-the-Art Machine Learning and Deep Learning Algorithms," IEEE Access, vol. 10, no. 14, pp. 39700–39715, 2022, doi: 10.1109/ACCESS.2022.3166891.

[2] J. I.-Z. Chen and K.-L. Lai, "Deep Convolution Neural Network Model for Credit-Card Fraud Detection and Alert," June 2021, vol. 3, no. 2, pp. 101–112, 2021, doi: 10.36548/jaicn.2021.2.003.

[3] M. Grant, "The 2017 Global Online Consumer Report: Key Figures in E-Commerce Worldwide," www.ecommerce-nation.com. 2017. [Online]. Available: https://www.ecommerce-nation.com/2017-global-online-consumer-report/

[4] R. Colaço and J. de Abreu e Silva, "Exploring the e-shopping geography of Lisbon: Assessing online shopping adoption for retail purchases and food deliveries using a 7-day shopping survey," J. Retail. Consum. Serv., vol. 65, no. 11, p. 102859, 2021, doi: 10.1016/j.jretconser.2021.102859.

[5] S. Kumar, "An Augmentation of Credit Card Fraud Detection using Random Undersampling," Turkish Online J. Qual. Inq., vol. 12, no. 6, pp. 664–674, 2023, [Online]. Available: https://www.tojqi.net/index.php/journal/article/view/1200

[6] I. Mekterović, M. Karan, D. Pintar, and L. Brkić, "Credit Card Fraud Detection in Card-Not-Present Transactions: Where to Invest?," Appl. Sci., vol. 11, no. 15, p. 6766, 2021, doi: 10.3390/app11156766.

[7] K. Patel, "Credit Card Analytics: A Review of Fraud Detection and Risk Assessment Techniques," Int. J. Comput. Trends Technol., vol. 71, no. 10, pp. 69–79, 2023, doi: 10.14445/22312803/ijctt-v71i10p109.

[8] J. Zheng, L. Yang, D. Xin, and M. Tian, "The Credit Card Anti-fraud Detection Model in the Context of Dynamic Integration Selection Algorithm," Front. Comput. Intell. Syst., vol. 6, no. 3, pp. 119–122, 2024, doi: 10.54097/a5jafgdv.

[9] U. Porwal and S. Mukund, "Credit Card Fraud Detection in E-Commerce," IEEE Xplore. pp. 280–287, 2019. doi: 10.1109/TrustCom/BigDataSE.2019.00045.

[10] Ghosh and Reilly, "Credit card fraud detection with a neural-network," IEEE Xplore, vol. 3. pp. 621–630, 1994. doi: 10.1109/HICSS.1994.323314.

[11] J. Akhilomen, "Data Mining Application for Cyber Credit-Card Fraud Detection System," Adv. Data Mining. Appl. Theor. Asp., vol. 13, no. 11, pp. 218–228, 2013, doi: 10.1007/978-3-642-39736-3_17.

[12] P. K. Chan, W. Fan, A. L. Prodromidis, and S. J. Stolfo, "Distributed data mining in credit card fraud detection," IEEE Intell. Syst., vol. 14, no. 6, pp. 67–74, 1999, doi: 10.1109/5254.809570.

[13] A. Srivastava, A. Kundu, S. Sural, and A. K. Majumdar, "Credit Card Fraud Detection Using Hidden Markov Model," IEEE Trans. Dependable Secur. Comput., vol. 5, no. 1, pp. 37–48, 2008, doi: 10.1109/tdsc.2007.70228.

[14] S. Maes, K. Tuyls, B. Vanschoenwinkel, and B. Manderick, "Credit Card Fraud Detection Using Bayesian and Neural Networks," Proc. First Int. NAISO Congr. NEURO FUZZY THECHNOLOGIES January 16 - 19, 2002 (Havana, Cuba., vol. 13, no. 11, 2002, [Online]. Available: https://researchportal.vub.be/en/publications/credit-card-fraud-detection-using-bayesian-and-neural-networks

[15] U. Nzenwata, O. Bakare, and O. Ukandu, "Artificial Intelligence: Positive or Negative Innovation," Int. J. Emerg. Trends Eng. Res., vol. 11, no. 7, pp. 245–252, 2023, doi: 10.30534/ijeter/2023/031172023.