International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-12 Issue-4, April 2023

Credit Card Fraud Detection using Machine Learning

Sagar Yeruva, Machavolu Sri Harshitha, Miriyala Kavya, Murakonda Sai Deepa Sree, Tumpudi Sri Sahithi

Abstract: ***Evolving technologies make human life easier with increasing challenges. Online payments have become an integral part of our lives in the era of digitalization. The credit card payment system has made transactions hassle-free. This led to E-Commerce appraisal. Digitalization of transactions has given rise to new forms of fraud and cyberattacks that can affect individuals and organizations. This had set hackers at a great deal to steal the cardholder details using different schemes. Credit card companies must recognize these fraudulent transactions at the earliest to retain credibility among the stakeholders. Traditional methods of fraud detection have proven ineffective in identifying and preventing these fraudulent activities and cyberattacks in real time. This paper discusses various Machine Learning algorithms that predict fraudulent transactions in real-time. Fraudulent activities are solved using data science and machine learning techniques with substantial processing power and the capacity to manage massive datasets. The model is trained on large volumes of the dataset. This paper emphasizes comparison of various machine learning algorithms' performance over the input. The accuracy and efficiency of several machine learning algorithms are measured and analyzed through tabulation and comparison. The trained model is integrated with a website to categorize financial transactions as either legitimate or fraudulent. On utilizing advanced machine learning algorithms, credit card fraud detection systems have become more refined and accurate in recent years. As a result, financial organizations and customers are protected against such fraudulent activities, leading to increased trust and confidence in utilization credit card payments.***

Keywords: ***Credit card fraud, Data Science, Fraudulent Transaction, Machine learning.***

Manuscript received on 19 February 2023 | Revised Manuscript received on 08 March 2023 | Manuscript Accepted on 15 April 2023 | Manuscript published on 30 April 2023.

*Correspondence Author(s)

Sagar Yeruva*, Department of CSE (AIML and IoT), VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad (Telangana), India. E-mail: sagar_y@vnrvjiet.in, ORCID ID: https://orcid.org/0000-0001-7467-8648

Machavolu Sri Harshitha, Department of CSE (AIML and IoT), VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad (Telangana), India. E-mail: harshimachavolu27@gmail.com, ORCID ID: https://orcid.org/0009-0008-8616-4327

Miriyala Kavya, Department of CSE (AIML and IoT), VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad (Telangana), India. E-mail: kavyamiriyala365@gmail.com, ORCID ID: https://orcid.org/0009-0005-0119-7663

Murakonda Sai Deepa Sree, Department of CSE (AIML and IoT), VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad (Telangana), India. E-mail: saideepasree03@gmail.com, ORCID ID: https://orcid.org/0009-0002-2041-7594

Tumpudi Sri Sahithi, Department of CSE (AIML and IoT), VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad (Telangana), India. E-mail: sahithi.tumpudi@gmail.com, ORCID ID: https://orcid.org/0009-0007-8587-7475

I. INTRODUCTION

Online transactions have been playing a crucial role in the modern era. A Credit card is a variety of payment cards where charges are created against a line of credit rather than the account holder's money deposits. Credit cards are dominant instruments for payment as they permit one to simply avail of an immediate line of short-run credit during transactions. Despite the merits of electronic payment systems, credit card organizations have been encountering growth in credit card fraud with the emergence of advanced technologies.

A bank is a financial organization that allows customers to make investments and deposit money. Fraud is an unlawful act with the objective of delusion to earn financially with no notice from the cardholder. On exposure to several frauds, one acquires crucial incompetence. Credit card fraud is the unofficial utilization of an account by somebody aside from the owner of that account. Credit cards are operated for the purchase of goods and services. Physical transactions are made by the insertion of a credit card into a machine at the merchant's store to get the products. This method might not be able to track transactions since the attacker had already stolen a credit card. By mode of online payment, attackers have little or no information about forged transactions.

Credit card fraud detection necessitates observance of the activities of users to classify and estimate objectionable behavior, that involves fraud. Necessary measures must be taken to eradicate such misemployment. The strategic response to fraudulent and illegal practices must be studied to reduce them. This is a significant drawback that highlights the demand for technologies like machine learning and data science which facilitate the reduction of fraudulent transactions. Classification and Identification of transactions as fraudulent turn out to be a tedious task. "According to the NCRB Report 2020, Debit, Credit card fraud online climbs steeply, 225% more cases when compared to 2019" [1]. Owing to the rapid-emerging online banking activity, over 40% of respondents stated that fraud would be their institution's highest priority in the year 2022 [2].

A. Credit Card Frauds

The quick financial world operates in a split second. All transactions, including withdrawals, deposits, and purchases, happen instantly. The Association of Certified defines fraud as any intentional or purposeful conduct that deprives another person of property or money by cunning, deception, or other unfair means. Credit card fraud can be identified as the following.

Point-of-Sale Fraud: Small skimming devices are affixed to standard Point-of-Sale equipment to hack the data. While the consumer completes the swipe transaction, these devices scan and store card information. This denotes a vendor giving this information with malice.

Skimming: This kind of credit card fraud involves the theft of credit card data using a small electronic tool called a "skimmer." The credit card is swiped through a "skimmer" that reads the magnetic strip on the card and records and saves the data. The blank magnetic strip of the credit card can then be written with the same information again [3].

Phishing: Phishing is a credit card fraud in which the cardholder receives an email claiming to be from a bank or financial institution. The cardholder is routed to a deceptive website that requests personal information after hitting the link in the email. The reason most individuals fall for this fraud is that the URL they receive in the email looks genuine.

Lost/Stolen Cards This kind of fraud happens when the card is misplaced by the user or stolen. The cardholder discovers the purchases through the monthly statement after the fraudster makes purchases using the card's data.

Account Takeover: Fraudsters gain access to the user's information and contact the credit card company to update the address information. The fraudster then receives a new card while posing as a legitimate user and reaps the benefits.

CNP (Card Not Present): The card may be used for benefits without the fraudster possessing it if they are aware of the card's expiration date and account information.

Counterfeit Card Fraud: Skimming is a common method for attempting this. A fake magnetic card is created using the exact specifications of the actual card. The fake card is functioning and may be used to make purchases.

ID Card Theft: This fraud is comparable to application fraud. To use the card or start a new account, the fraudster needs the original card information, which is obtained via ID theft. The most challenging fraud to spot is this kind.

B. History of Credit Card Frauds

Credit card fraud is known as theft due to unauthorized access to credit card and account information. Stolen, misplaced, or counterfeit credit cards are often stolen. With the development of e-commerce, various frauds such as the non-existence of cards have occurred, and the possibility of fraud increases in e-commerce.

In this digital world, all individuals need to be aware of credit card fraud to protect themselves from scammers. It is also important to develop good money habits, such as throwing your credit card out of the ATM.

India ranks among the top five countries for credit card fraud, with more than a third of people saying they have been swindled.

By 2025, the total global payment card volume is projected to reach US$56.182 trillion (about $170,000 per person in the US), while the total value of card fraud worldwide is projected to reach US$35.31 billion (about $110 per person in the US) (about $110 per person in the US). Fraud dropped to 6.28 cents per $100 gross. US Fraud Losses are Estimated at $12.

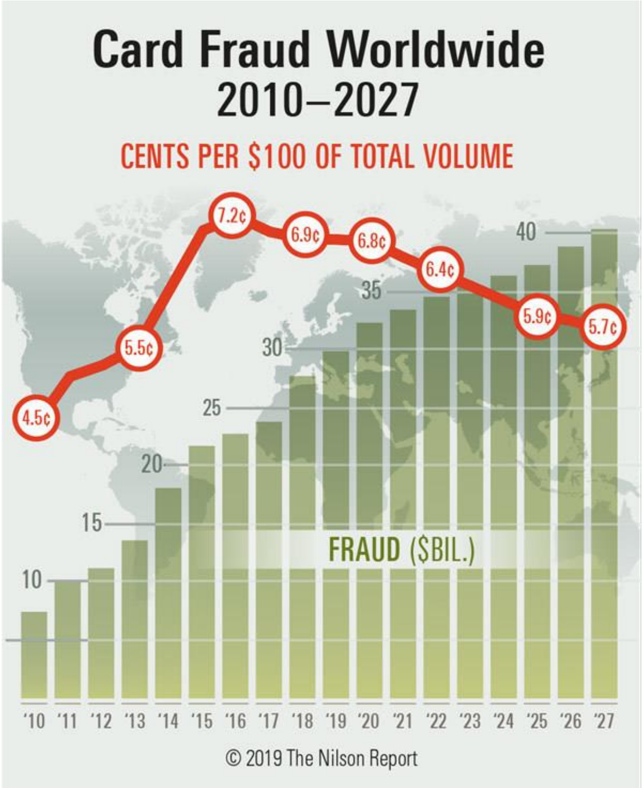

Fig 1: Nilson Report credit card fraud analysis, 2010-2027.

II. LITERATURE SURVEY

A. Detecting Credit Card Fraud Using Machine Learning and Data Science [4]

Arjwan H. Almuteer proposed a system to detect credit card fraud using Local Outlier Factors and Isolation Forest. The accuracy obtained is 99.6%. Considering only one-tenth of the data available, the precision reaches only up to 28%.

B. Credit Card Fraud Detection using Machine Learning Algorithms [5]

Vaishnavi Nath and Geetha S presented a system that concentrates on a few instances of fraud that occur in real-world transactions. To tackle each instance of fraud, we employed a sequence of machine learning models and determined the optimal approach through evaluation. Additionally, we evaluated a novel approach that effectively deals with imbalanced data distribution. Our experiments employed data from a financial institution under a confidential disclosure agreement.

C. Application of Hidden Markov Model in Credit Card Fraud Detection [6]

Bhusari and S. Patil comparatively analyzed, and examined the efficacy of naive Bayes, k-nearest neighbor, and logistic regression models on credit card fraud data that was highly skewed. The dataset was sourced from European cardholders and comprised 284,807 transactions. The models' effectiveness was evaluated using metrics such as accuracy, sensitivity, specificity, precision, Matthew's correlation coefficient, and balanced classification rate.

Retrieval Number:100.1/ijeat.D40480412423
DOI: 10.35940/ijeat.D4048.0412423
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

D. Data Mining Application in Credit Card Detection system [7]

Francisca Nonyelum Ogwueleka discusses classification based on supervised learning. The dataset utilized for the classification contains 30 features, but to maintain confidentiality, all but 28 of them were transformed using Principal Component Analysis and had unknown labels. Upon pre-processing the dataset with normalization and Principal Component Analysis, all classifiers were able to achieve an accuracy of over 95.0%.

III. ABOUT THE DATASET

It is possible to classify transactions into two categories, fraudulent transactions, and valid transactions. Dataset source: Kaggle. This dataset is sourced by an unnamed institute. This dataset has 8 columns. There are 1000000 entries. Out of them, 912,597 are classified as valid transactions and 87,403 as fraudulent transactions. They are

- Distance from home,
- Distance from the last transaction,
- Ratio to the median purchase price,
- Repeat retailer
- Used chip,
- Used PIN,
- Online order,
- Fraud

Insights from the study of the dataset:

- A transaction is said to be fraudulent if the distance of the last transaction is in the range of 0 – 0.5 in the case of offline transactions.
- Fraudulent transactions possess a PIN field value of 0 (pin is not used).
- If the Ratio to the median purchase price is low, but the distance from the last transaction is in the hundreds, it is considered a fraudulent transaction.
- Transactions are considered fraudulent when repeated with retailers.
- A valid transaction is when the ratio to the median purchase price is less with a higher value of the distance from the last transaction.

IV. METHODOLOGY

Many algorithms can be used to detect whether a transaction is fraudulent or not. We have considered a few algorithms – Logistic Regression, XGBoost, K-Nearest Neighbors, and Decision Trees.

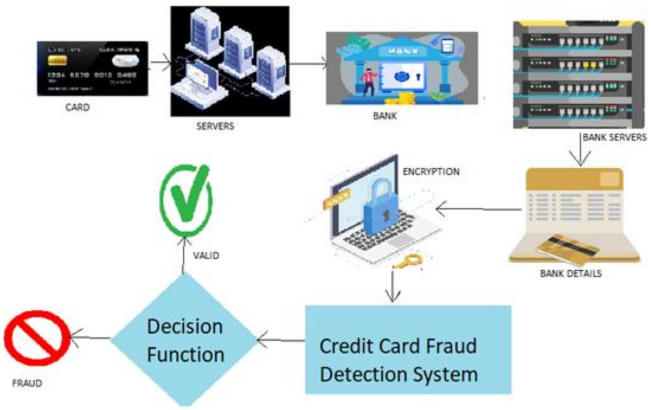

Fig 2: System Architecture

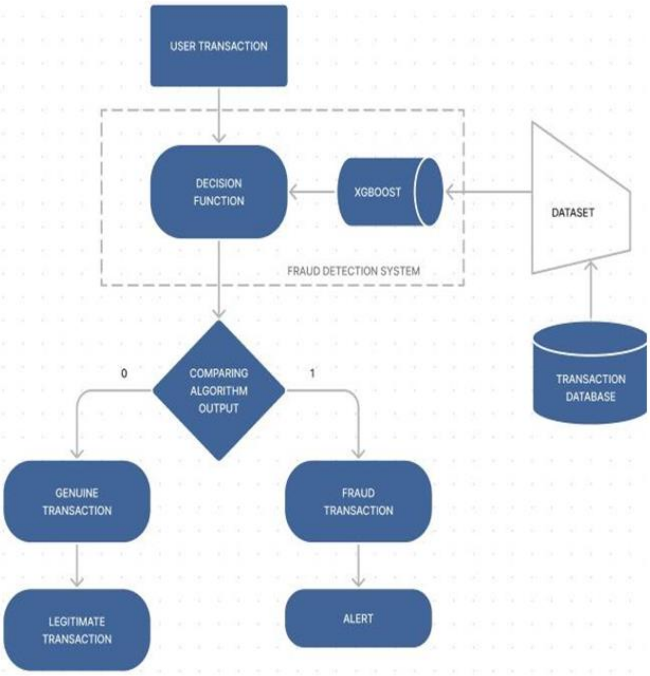

Fig 3: Block Diagram of Credit Card Fraud Detection

A. Logistic Regression

The logistic model is a statistical model that models the probability of an event taking place by having the log odds for the event be a linear combination of one or more independent variables. Logistic regression can also be defined in terms of regression analysis. In regression analysis, logistic regression (or logit regression) is estimating the parameters of a logistic model. The corresponding probability of the value labeled can vary between 0 and 1. Hence the labeling; the logistic function converts log odds to probability.

Logit(pi) = 1/ (1+ exp(-pi))

ln(pi/(1-pi)) = Beta_0 + Beta_1*X_1 + … + B_k*K_k

where, logit(pi) is the response or dependent variable and x is the independent variable. In this model, the beta parameter, or coefficient is commonly estimated via maximum likelihood estimation (MLE).

B. XGBoost

Extreme Gradient Boosting (XGBoost) a C++ library efficiently optimizes the training for Gradient Boosting. XGBoost is an implementation of Gradient Boosted decision trees. Gradient boosting is a supervised learning algorithm that accurately predicts a target variable by combining an ensemble of estimates from a set of simpler and weaker models. In this algorithm, decision trees are created in sequential form. Weights play a significant role in XGBoost. Independent variables assigned by weights are then fed into the decision tree to predict results. The weight of variables predicted wrong by the tree is increased and then fed to the second decision tree. These individual classifiers/predictors then ensemble to give a strong and more precise model. Regression, classification, ranking, and user-defined prediction problems can be predicted by this algorithm.

Retrieval Number:100.1/ijeat.D40480412423
DOI: 10.35940/ijeat.D4048.0412423
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

C. K Nearest Neighbors

The K-Nearest Neighbors' classifier is a non-parametric supervised machine learning algorithm. It is distance-based: it also classifies objects based on their proximate neighbor's classes. K defines the number of nearby observations to use in the algorithm. It is the number of nearest neighbors to use. For classification, a majority vote is used to determine which class a new observation should fall into. Larger values of K produce more stable decision boundaries are often and more robust to outliers than exceedingly small values (K=5 would be better than K=1 to predict desirable results.)

D. Decision Trees

A Decision Tree is a Supervised learning technique that is a tree-structured classifier, the features of a dataset are represented by internal nodes, decision rules by branches, and the outcome by leaf nodes. This can be used for both classification and Regression problems, but mostly used for Classification problems.

V. RESULTS AND DISCUSSION

The model is developed in a python environment and three algorithms like Logistic Regression, XGBoost, KNN, and Decision Tree were experimented as shown in Fig 4.1, Fig 4.2, Fig 4.3, and Fig 4.4 using the dataset described above and can observed the performance of each algorithm about confusion matrix and can observe the accuracy of each algorithm in Table-1. The following Table-1 presents the accuracy of various algorithms in various phases of the model building like training, validation, and testing.

Table-I: Train, Validation, and Test the Accuracy of four algorithms – Logistic Regression, XGBoost, KNN, and Decision Trees.

Model Name	Accuracy during Training	Accuracy during Testing	Accuracy during Validation
Logistic Regression	95.87%	95.86%	48.12%
XGBoost	100%	99.99%	99.98%
KNN	98.85%	98.21%	75.62%
Decision Tree	99.85%	99.85%	98.09%

The implementations of all the algorithms are shown in the given figures below.

(i) Logistic Regression

```
In [13]: X = data.iloc [:, :-1]
         Y = data.iloc[:,-1]
         X_train, X_test, y_train, y_test = train_test_split(X,Y, test_size = 0.30, random_state = 1)

In [14]: lr = LogisticRegression(max_iter  =10000)
         %time model = lr.fit(X_train, y_train)
         scores = cross_val_score(model,X_train,y_train,scoring = 'r2',cv = 5)
         scores
         lr_pred = lr.predict(X_test)
         TrainAccuracy =  accuracy_score(y_train, model.predict(X_train))
         TestAccuracy =  accuracy_score(y_test, lr_pred)
         ConfusionMatrix =  confusion_matrix(y_test, lr_pred)
         print("Accuracy of validation data is {}".format(scores.mean()))
         print('Accuracy score of the train data is {}'.format(TrainAccuracy))
         print('Accuracy score of the test data is {}'.format(TestAccuracy))
         print('Confusion Matrix - {}'.format(ConfusionMatrix))

         CPU times: total: 7.19 s
         Wall time: 9.14 s
         Accuracy of validation data is 0.4812050367132478
         Accuracy score of the train data is 0.9587671428571428
         Accuracy score of the test data is 0.95862
         Confusion Matrix - [[271790   1873]
          [ 10541  15796]]
```

Fig 4.1: Logistic Regression Implementation

(ii) XGBoost

```
In [15]: from xgboost import XGBClassifier
         param = {'learning_rate': 0.2,
                  'max_depth': 2,
                  'n_estimators':200,
                  'subsample':0.9,
                 'objective':'binary:logistic'}
         xgb_model = XGBClassifier()
         %time xgb = xgb_model.fit(X_train, y_train)
         # y_test_pred_proba = xgb.predict_proba(X_test)[:,1]
         # roc_auc = metrics.roc_auc_score(y_test, y_test_pred_proba)
         xgb_scores = cross_val_score(xgb,X_train,y_train,scoring = 'r2',cv = 5)
         xgb_scores
         xgb_pred = xgb.predict(X_test)
         xgb_TrainAccuracy =  accuracy_score(y_train, xgb.predict(X_train))
         xgb_TestAccuracy =  accuracy_score(y_test, xgb_pred)
         xgb_ConfusionMatrix =  confusion_matrix(y_test, xgb_pred)
         print("Accuracy of validation data is {}".format(xgb_scores.mean()))
         print('Accuracy score of the train data is {}'.format(xgb_TrainAccuracy))
         print('Accuracy score of the test data is {}'.format(xgb_TestAccuracy))
         print('Confusion Matrix - {}'.format(xgb_ConfusionMatrix))

         CPU times: total: 2min 45s
         Wall time: 31.1 s
         Accuracy of validation data is 0.9998026505875284
         Accuracy score of the train data is 1.0
         Accuracy score of the test data is 0.99998
         Confusion Matrix - [[273662      1]
          [     5  26332]]
```

Fig 4.2: XGBoost Implementation

(iii) K Nearest Neighbours

```
In [16]: n = 7
         KNN = KNeighborsClassifier(n_neighbors = n)
         %time knn = KNN.fit(X_train, y_train)
         knn_pred = knn.predict(X_test)
         knn_scores = cross_val_score(knn,X_train,y_train,scoring = 'r2',cv = 5)
         knn_scores
         knn_TrainAccuracy =  accuracy_score(y_train, knn.predict(X_train))
         knn_TestAccuracy =  accuracy_score(y_test, knn_pred)
         knn_ConfusionMatrix =  confusion_matrix(y_test, knn_pred)
         print("Accuracy of validation data is {}".format(knn_scores.mean()))
         print('Accuracy score of the train data is {}'.format(knn_TrainAccuracy))
         print('Accuracy score of the test data is {}'.format(knn_TestAccuracy))
         print('Confusion Matrix - {}'.format(knn_ConfusionMatrix))

         CPU times: total: 1.64 s
         Wall time: 3.45 s
         Accuracy of validation data is 0.7562558294291472
         Accuracy score of the train data is 0.9885357142857143
         Accuracy score of the test data is 0.98213
         Confusion Matrix - [[270476   3187]
          [  2174  24163]]
```

Fig 4.3: KNN Implementation

(iv) Decision Trees

```
In [17]: # Model with optimal hyperparameters
         decision_tree_model = DecisionTreeClassifier(criterion = "gini",
                                         random_state = 100,
                                         max_depth=5,
                                         min_samples_leaf=100,
                                         min_samples_split=100)

         %time dt = decision_tree_model.fit(X_train, y_train)
         dt_pred = dt.predict(X_test)
         dt_scores = cross_val_score(dt,X_train,y_train,scoring = 'r2',cv = 5)
         dt_scores
         dt_TrainAccuracy =  accuracy_score(y_train, dt.predict(X_train))
         dt_TestAccuracy =  accuracy_score(y_test, dt_pred)
         dt_ConfusionMatrix =  confusion_matrix(y_test, dt_pred)
         print("Accuracy of validation data is {}".format(dt_scores.mean()))
         print('Accuracy score of the train data is {}'.format(dt_TrainAccuracy))
         print('Accuracy score of the test data is {}'.format(dt_TestAccuracy))
         print('Confusion Matrix - {}'.format(dt_ConfusionMatrix))

         CPU times: total: 1.45 s
         Wall time: 3.05 s
         Accuracy of validation data is 0.9809647549021205
         Accuracy score of the train data is 0.9985757142857142
         Accuracy score of the test data is 0.9985
         Confusion Matrix - [[273320    343]
          [   107  26230]]
```

Fig 4.4: Decision Trees Implementation

Considering the XGBoost algorithm, it is having highest validation accuracy and highest test accuracy. The Confusion matrix derived also tells that the trained model of the XGBoost algorithm has extremely low false negatives and false positives compared to true positives and true negatives which makes this model much better than other considered models. So, considering the XGBoost algorithm will help to detect fraudulent transactions very accurately.

The outputs given by the python code written to predict whether given transactions are fraudulent are:

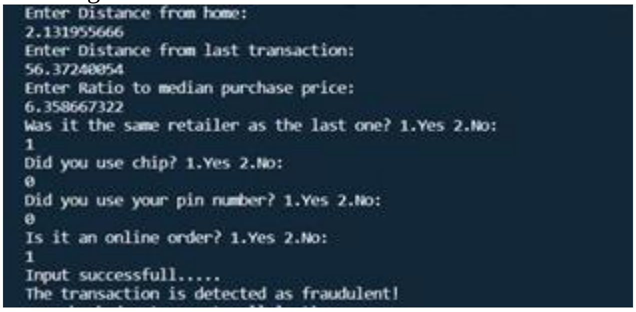

Fig 5.1: Output from the prototype

Fig 5.2: Output from the prototype

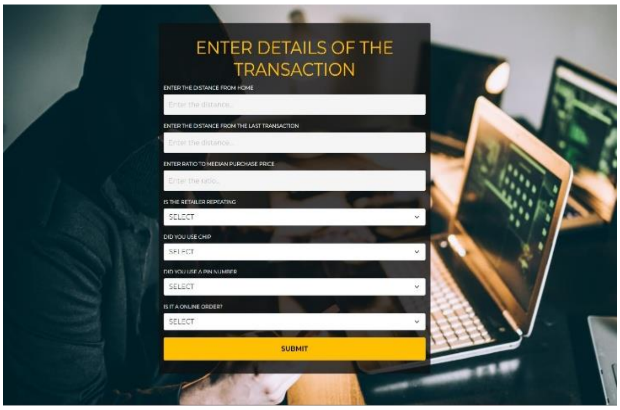

Fig 5.3: Application Interface

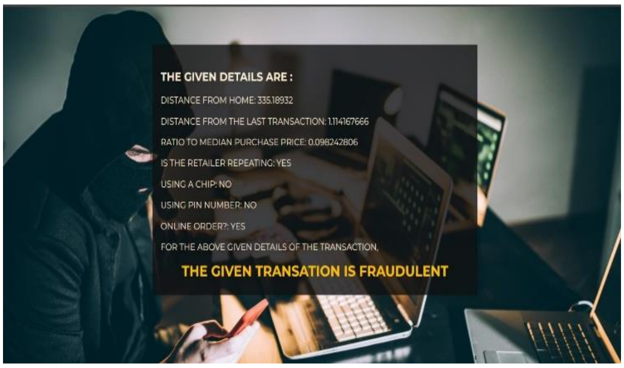

Fig 5.4: Transaction Result

The model is also supported with an application interface that could help any stakeholder to make use of the model using a web application. Fig 5.1 and Fig 5.2, present the working model upon submission of the necessary input data elements and the detection of the model in terms of a fraudulent transaction. As shown Fig 5.3 and Fig 5.4 presents the final working model of the application through the web application interface.

DECLARATION

Funding/ Grants/ Financial Support	No, I did not receive.
Conflicts of Interest/ Competing Interests	No conflicts of interest to the best of our knowledge.
Ethical Approval and Consent to Participate	No, the article does not require ethical approval and consent to participate with evidence.
Availability of Data and Material/ Data Access Statement	Not relevant.
Authors Contributions	All authors have equally participated in various phases of the work like identification of the problem, review of literature, design of the problem, development of the work, testing and deployment of the model in the web application.

VI. CONCLUSION

Considering all the results of various Machine Learning algorithms, the most accurate and effective algorithm that best suits our current prototype is selected. XGBoost algorithm has given better results in terms of accuracy. After the ML (Machine Learning) model is trained with the considered dataset, and the model is done with proper testing, an input set consisting of features of the transaction is given to the model. The model then gives out the output predicting if the given input transaction is fraudulent.

REFERENCES

1. https://www.prnewswire.com/news-releases/payment-card-fraud-losses-reach-27-85-billion-300963232.html
2. https://www.abrigo.com/blog/fraud-concerns-and-trends-in-2022/
3. https://www.idfcfirstbank.com/finfirst-blogs/credit-card/credit-card-fraud-in-india
4. S P, Maniraj & Saini, Aditya & Ahmed, Shadab & Sarkar, Swarna. (2019). Credit Card Fraud Detection using Machine Learning and Data Science. International Journal of Engineering Research and. 08. 10.17577/IJERTV8IS090031[CrossRef]
5. Vaishnavi Nath Dornadula, S Geetha, Credit Card Fraud Detection using Machine Learning Algorithms, Procedia Computer Science, Volume 165, 2019, Pages 631-641, ISSN 1877-0509 [CrossRef]
6. Bhusari, V & Patil, S. (2011). Application of Hidden Markov Model in Credit Card Fraud Detection. International Journal of Distributed and Parallel systems. 2. 10.5121/ijdps.2011.2618. [CrossRef]
7. Ogwueleka, Francisca Nonyelum. "Data mining application in credit card fraud detection system." Journal of Engineering Science and Technology 6.3 (2011): 311-322

AUTHORS PROFILE

Sagar Yeruva Associate Professor in the Department of CSE (AIML and IoT) in Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology (VNRVJIET), Hyderabad, India. He has a Ph.D. in CSE from Jawaharlal Nehru Technological University, Hyderabad, and has 19 Years of teaching experience. He has authored 30 research publications in various journals/conferences. He also acted as a review member for various journals and conferences. His areas of research include Data Bases, Data Mining, Machine Learning, and Distributed Computing.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Machavolu Sri Harshitha, Pursuing B. Tech in CSE (Internet of Things) from Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology (VNRVJIET), Hyderabad, India. Her research areas include Artificial Intelligence, Machine Learning, Internet of Things, and Programming concepts.

Miriyala Kavya Naga Durga Lakshmi, Pursuing B. Tech in CSE (Internet of Things) from Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology (VNRVJIET), Hyderabad, India. Her research areas include Artificial Intelligence, Machine Learning, Internet of Things, and Programming concepts.

Murakonda Sai Deepa Sree, Pursuing B. Tech in CSE (Internet of Things) from Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology (VNRVJIET), Hyderabad, India. Her research areas include Artificial Intelligence, Machine Learning, Internet of Things, and Programming concepts.

Tumpudi Sri Sahithi, Pursuing B. Tech in CSE (Internet of Things) from Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology (VNRVJIET), Hyderabad, India. Her research areas include Artificial Intelligence, Machine Learning, Internet of Things, and Programming concepts.

Retrieval Number:100.1/ijeat.D40480412423
DOI: 10.35940/ijeat.D4048.0412423
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)