WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

CREDIT CARD FRAUD DETECTION USING MACHINE LEARNING AND DATA SCIENCE

Akshada dighe[1], Shantanu Patil[2], Pratik Thakre[2], Gauri Tupe[2], Pranav Kenjale[2]

[1]Assistant Professor, Department of Information Technology, Genba sopanrao moze college of engineering
[2]BE IT Students, Genba sopanrao moze college of engineering

ABSTRACT

In today's lucrative scenario, credit card use has become extremely predominant. It is crucial that credit card companies are capable of recognizing fraudulent credit card transactions so that clients are not levied for items that they did not buy. Such problems can be tackled with Data Science and its significance, along with Machine Learning, cannot be dramatized. This project aims to illustrate the modeling of a data set using machine learning with Credit Card Fraud Detection. The Credit Card Fraud Detection Problem involves modeling past credit card transactions with the data of those that turned out to be a fraud.

This model is then used to realize whether a new transaction is fraudulent or genuine. Machine learning algorithms analyze all the authorized transactions and report the doubtful ones. These reports are explored by professionals who contact the cardholders to confirm if the transaction was genuine or fraudulent. Our goal here is to detect 100% of the fraudulent transactions while reducing the incorrect fraud classifications. Credit Card Fraud Detection is an example of classification that comes under supervised learning. In this process, we have concentrated on analyzing and pre-processing data sets as well as the deployment of multiple anomaly detection algorithms such as Random Forest Classifier, AdaBoost Classifier, CatBoost Classifier, and XGBoost Classifier. Scikit-learn is chiefly written in Python and uses NumPy broadly for speedy linear algebra and array operations. In this project, we have used different algorithms implemented in sklearn library.

Keywords*: Credit Card Fraud, Machine Learning, Scikit-Learn, Random Forest Algorithm, AdaBoost Algorithm, CatBoost Algorithm, XGBoost Algorithm.*

1. INTRODUCTION

In today's lucrative scenario economic losses are gaining speed. There are hackers all around the world. Innocent people are being cheated. There are sometimes when it's very difficult to make out is it a fraud or a genuine record. The data scientists are continuously finding patterns that will give the idea about fraud and genuine records/ transactions. Credit card fraud costs customers and monetary companies billions of dollars annually, and the fraudsters constantly try to find new rules and tactics to commit illegal actions. Thus, fraud detection systems have become vital for banks and financial institutions, to reduce their losses. However, there is an inadequacy of published literature on credit card fraud detection systems, due to limited credit card transactions dataset for scientists. The most commonly used fraud detection methods are Decision Trees, Logistic Regression, and Random Forest classifiers. But amid all existing classifiers, boosting classifiers is recognized as a popular and common method, not because of their quite uncomplicated implementation, but also due to its extraordinary predictive execution on practical problems. Around the world today as every transaction is taking place online and there's less of a physical presence, these frauds have increased drastically. We can consider two types of fraud one in which a credit card is present and the other in which the card is not present. If we ponder over this, we get to know the first type of fraud is very rare or not so common these days but the second one is accelerating. The reason being is, if the physical card was stolen or looted, the authorized person could easily report that issue. On the other hand, if the card holder's details like account details were leaked or misplaced and the fraudster kept it with him for months then it becomes very tedious to make out the source of compromise. The cardholder might be unaware of this until he receives the statement. For this, the cardholders must continuously check their accounts for any fraudulent or unknowing transactions that happened. These credit card frauds are just one type of fraud happening. There are n number of frauds going around in the world, for instance, cell phones, insurance claims, tax return claims, etc. A team of Data Scientists, Data Analysts are constantly working on them and finding out ways to discover frauds and keep the users safe from these losses. They are using techniques of Data mining, Data analysis, and machine learning for obtaining successful solutions to these problems. Mostly the internal control systems are weak and hence, we are using data science techniques to tackle this problem. They are using multiple machine learning, statistics, and artificial intelligence techniques. Let's discuss more datasets, pre-processing, processing, model training, and predictions, in upcoming paragraphs.

In our credit card fraud detection project, we have used a few the machine learning algorithms like bagging and boosting. Our main aim is to compare 4 different classifiers and find the fastest and most accurate one. Then we use this classifier at the backend of a flask application. So, basically, we have a web application that has an ML model running at the backend. This was a bit tricky but an interesting task as we explored the world of Python. Talking more about the dataset we used for training our models is from kaggle.com. This is the only dataset available so far. creditcard.csv has more

than 2 lakh transactions. As this data is collected from a bank, they cannot disclose customers' personal details. Hence the data is pre-processed. There are 31 columns amongst which V1 to V28 features are obtained as a result of Principal Component Analysis, and these are the encoded features of the customers. For our application, we have trained the best model with 2 columns i.e., independent variables namely Time and Amount and the output or independent variable is Class which contains 0 or 1. Class 0 represents that the transaction is genuine and 1 represents that it is a fraudulent transaction. The model studies the specific behavior and predicts an accurate result. The second most important component of our project is the machine learning model. We have demonstrated the use of 4 models out of which 1 is bagging and the rest 3 are boosting algorithms. The reason for using the 1 bagging algorithm is because the paper we referred to gave us a conclusion that Random Forest Classifier being an ensemble learning method is the most accurate. Yes, no doubt it is the most accurate one and its accuracy is the best amongst all others, but the second main aspect is time constraint which is not accepted at all to be high. The accuracy must be high but the time required should be minimum. When we compared the execution time between Random Forest and rest 3 boosting algorithms, we found it is the slowest of all. Our task is to produce an application that is fast and accurate. Hence, we have chosen Catboost Classifier as the best one because it is accurate and the time required is very less than compared RFC. The main aim of our project is to carry forward a research project using new algorithms which are one step ahead of the existing ones. During the course of this project, we learned different stages of data science and machine learning. These include data collection, managing incorrect/ wrong data, checking for missing values, sampling data for model training and predicting results. This project clearly shows which algorithm is the best and should be used for real-time fraud detection. On a final note, every dataset is different and so are the results, we can say that everything depends on the type of dataset we use. This is a project on Credit Card Fraud Detection which uses Catboost Classifier and gives accurate and fast results.

2. LITERATURE REVIEW

Many books have been published and made available to the public, either as fraudulent or proven to be fraudulent. Learn how to identify credit card fraud. Although these methods and algorithms have achieved unexpected success in some areas, they have not provided reliable and consistent fraud detection solutions.

This method has proven effective in reducing false positives and improving fraud detection performance. The author studies various methods of finding external objects. We use these methods to detect fraud. Credit card fraud can be addressed through these external methods. These methods are useful for finding fraudulent jobs.

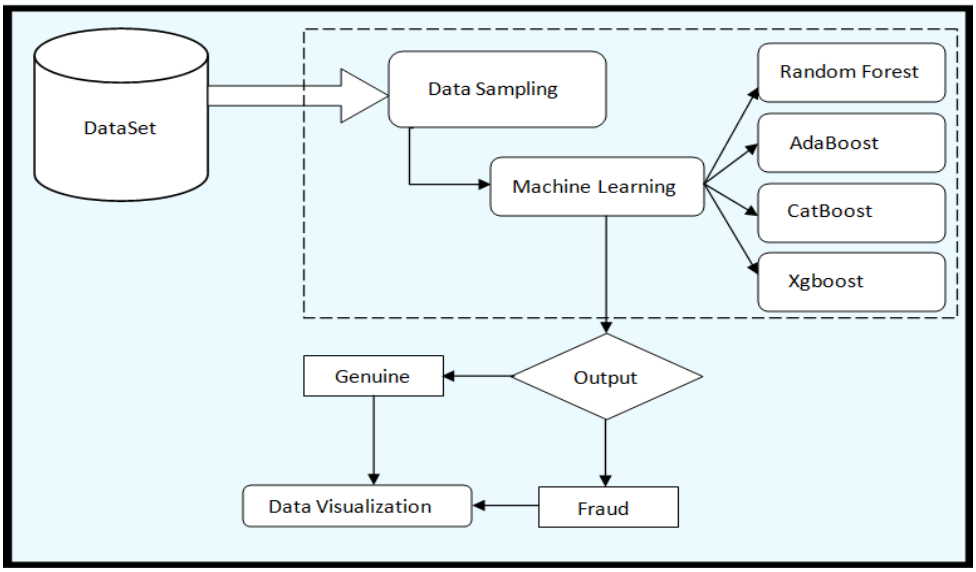
Three different models are supported in neural networks and the proximity of the nearest logistic neighbor is improved. To test these examples, 70% of the area is used for training and 30% is used. Set aside for review and testing. Accuracy, Sensitivity, Clarity, Accuracy, Matthew Correlation Coefficient and Balance. The degree of differentiation is used to measure performance in three categories.

The nearest anti-k neighbor data propagation detection algorithm for credit card fraud detection, whereas traditional methods require multiple sites crawls to get unused fraud data circulating in the stream. This makes it easy to prevent fraudulent activity and credit card fraud. Sequence number checking and error checking is also simple and easy to find bad numbers.

An increase in Internet activity is directly related to an increase in the level of fraud. In this article, various algorithms such as KNearest Neighbor, Rainforest, AdaBoost, cat boost, Xgboost, and logistic regression are some of the common separations of task problems, and the moral appears to be very similar. Limit work time, amount, and frequency to obtain these features. In this article, KNearest's four neighbors are different algorithms called AdaBoost, Rainforest, and Logistic Reduction compared to fraud detection methods. Reduced efficiency of some algorithms

3. PROPOSED METHODOLOGY

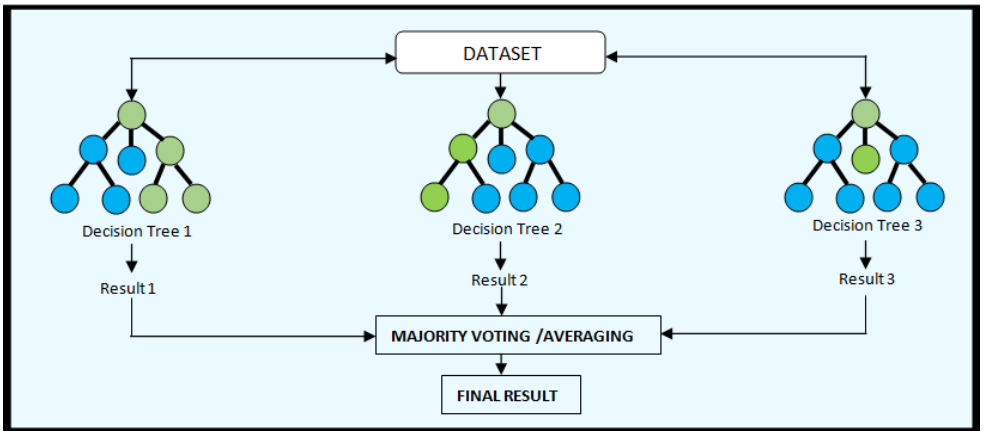
It is always difficult to detect fraud transactions based on previous transaction data using existing methods, and it is difficult to track because 99% of users do not report fraud. Traditional methods that have long been used to detect fraud transactions are time-consuming and inadequate to detect fraud transactions. We need a system that can detect fraudulent transactions by analyzing previous data. The method proposed in this white paper uses modern machine learning algorithms to detect missing quantities and external factors. Our model uses early data processing techniques with feature selection and credit card fraud set size minimization to reduce the number of input features before inclusion in the model. The student's continuous short-term memory is then used as a flexible pattern recognition component to capture the continuous dependencies between credit card purchases. Next, a high-profile approach is introduced; with a unique focus on the information that emerges from the hidden layers of short-term memory, the model finds the fraudulent patterns and finds the best jobs that vary greatly depending on the consumer's purchases. I made it possible. The proposed system uses a variety of machine learning algorithms to analyze the data and obtain the desired output.



(Fig 1. System Design)

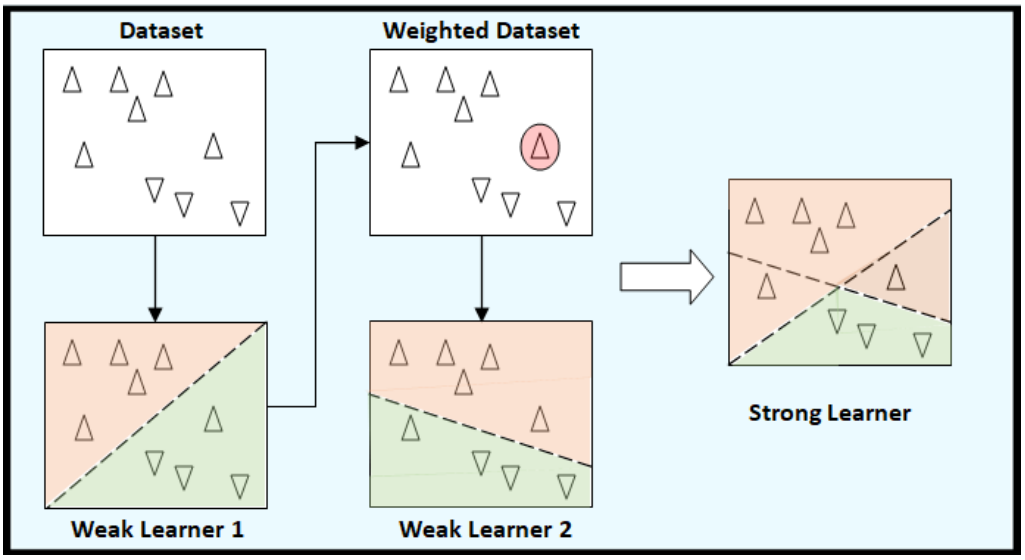
This system consists of four algorithms: one for bagging and the other three for boosting.

1) Random forest algorithm: Random forest is a supervised machine learning algorithm commonly used for classification and regression problems. Create a decision tree from different samples and make a majority vote to rank and average in the event of a recession. One of the most important features of the Random Forest algorithm is the ability to manage datasets containing continuous variables, as in the case of regression and categorical variables. You will get the best results for classification problems.



(Fig 2. Random Forest Algorithm)

2) Adaboost: AdaBoost is the most representative algorithm in the boosting family. This saves the distribution of a series of opportunities for training the samples and corrects this distribution of opportunities for each sample during each multiplication. By reading the algorithm directly, we generate an element separator and calculate its error rate in the training example. AdaBoost uses this error rate to modify the probability distribution of the training pattern. The role of weight change is an important function of misweighed samples and will reduce weight if the samples are properly classified. Finally, a powerful phase is established by weighted voting by individual separatists.



(Fig 3. Adaboost Classifier)

3) Catboost: CatBoost is an algorithm for increasing the gradient of a decision tree. Developed by Yandex researchers and engineers, this algorithm follows the widely used MatrixNet algorithm for measuring, predicting, and recommending tasks in the enterprise. It is available worldwide and can be used in many different disciplines and issues. Catboost gets the best benchmark results and that's great.

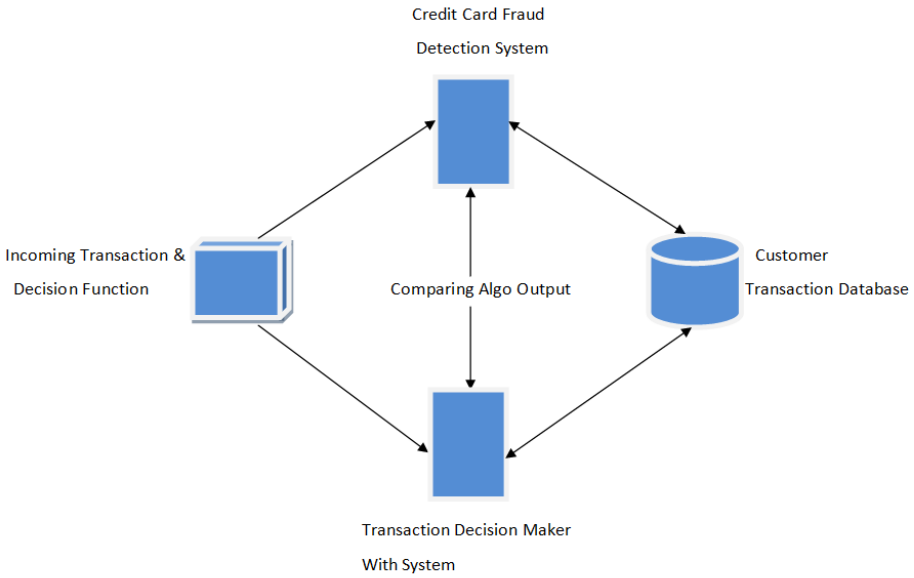
4) Xgboost: With this algorithm, decision trees are built one after another. Weight plays an important role in XGBoost. All independent variables are weighted and included in the decision tree that predicts the outcome. The weights of variables that are predicted to be wrong in the tree are increased, and these variables are sent to the second decision tree. Then compile these individual classifiers/predictors to provide a more robust and accurate model. It can address user-defined retransmission, classification, level, and rate issues.

Boosting Methods	Accuracy	Time
AdaboostClassifier	98.3033	156
XGBoostClassifier	98.9526	123
RandomForestClassifier	99.9894	460
CatBoostClassifier	99.9529	152

(Fig 3. Comparison of Classifiers)

CatBoost Classifier is the most accurate classifier among the above four algorithms

Implementation:



(Fig. 4 System Overview of the Credit Card Fraud Detection System)

The proposed model for Credit Card Fraud Detection is depicted in the figure above. The steps that are undertaken to develop the proposed model are explained below.

Step 1: Loading the DataSet: - We took our dataset from kaggle.com. This dataset consists of 2 lakhs transactions. As this data is collected from a bank, they cannot disclose customers' personal details. Hence the data is pre-processed. There are 31 columns amongst which V1 to V28 features are obtained as a result of Principal Component Analysis, and these are the encoded features of the customers.

```
In [8]: df=pd.read_csv('creditcard.csv')
```

Step 2.1: - Data Processing: - For our application we have trained the best model with 2 columns i.e., independent variables namely Time and Amount and the output or independent variable is Class which contains 0 or 1. Class 0 represents that the transaction is genuine and 1 represents that it is a fraudulent transaction. The model studies the specific behavior and predicts an accurate result.

Step 2.2: - Data Sampling: - Here in we have Over-Sampling and Under-Sampling. We have used Over-Sampling in our project. We could have used Under-Sampling but the data we have is highly imbalanced data giving us only 492 fraud transactions out of 2 lakh transactions. And Under-Sampling uses imitation to decrease the no. of high-class samples equal to the number of low-class samples and that would create a problem for prediction. Thus, we used OverSampling which increases the number of low-class samples equal to high-class samples for balancing with low-class. Lastly, our data is balanced using Over-Sampling.

Step 3.1: - Model Training: - Model validation is very important to ensure the verification of the model predictions. That's the reason we went for Training after balancing this data, we went for dividing the data into 2 parts namely, training data and testing data. The training data was used for training the four machine learning models which were around 75% of the total data available.

```
In [30]: from sklearn.ensemble import RandomForestClassifier
```

```
In [ ]: from sklearn.ensemble import AdaBoostClassifier
```

```
In [ ]: ada=AdaBoostClassifier()
        ada.fit(x_train,y_train)
```

```
In [ ]: from xgboost import XGBClassifier
```

XGBClassifier time :-> 58 sec

1. 58 s
2. 2 m 3 s

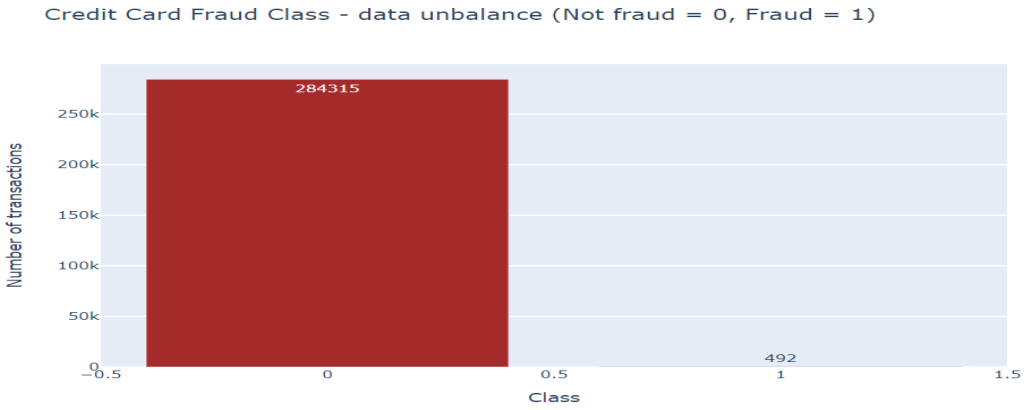
Step 3.2: - Model Testing: - The remaining 25% of data was used to test the trained models i.e., for prediction purposes.

Step 3.3: - Evaluation: - Our main aim was to compare the best model i.e., Random Forest Classifier which comes under ensemble learning methods and is a bagging method with other three boosting methods that included AdaBoost, CatBoost, and XGBoost Classifiers. The result was clear that Catboost gave us high accuracy in very less time.

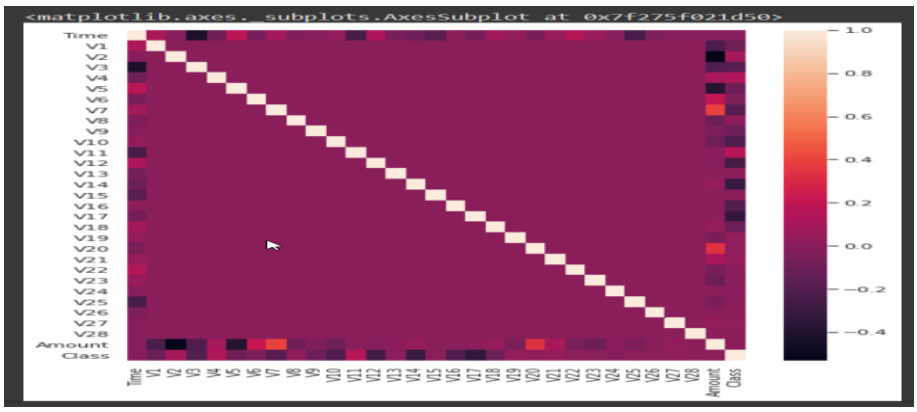
Step 4: Trained Model: - After comparing the classifiers we came to a conclusion stating that CatBoost Classifier is the best one among the other three classifiers. The result was clear that Catboost gave us high accuracy in very less time.

Step 5: - Fraud/ Genuine: - The model after getting the value predicts the transactions and gives an accurate result to the user whether it is Fraud or Genuine.

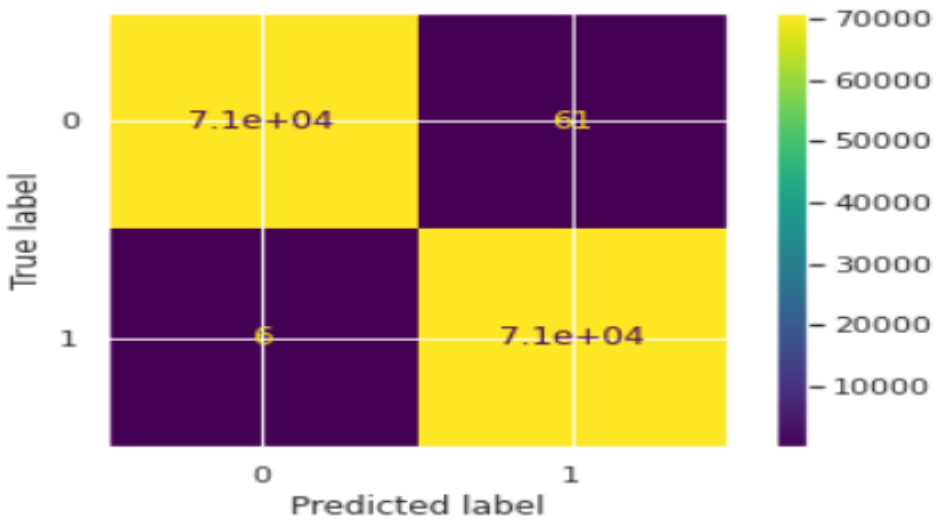
4. RESULTS



(Fig. 5 Bar Graph of Fraudulent and Genuine Transactions)



(Fig 6. Heatmap)



(Fig 7. Confusion matrix)

```
In [ ]: prediction_ada=ada.predict(x_test)
        print("Classification Report on Hold Out Dataset==>\n\n",metrics.classification_report(y_test,prediction_ada))
```

```
Classification Report on Hold Out Dataset==>

              precision    recall  f1-score   support

           0       0.98      0.99      0.98     71079
           1       0.99      0.98      0.98     71079

    accuracy                           0.98    142158
   macro avg       0.98      0.98      0.98    142158
weighted avg       0.98      0.98      0.98    142158
```

(Fig. 8 Classification Report on Hold out Dataset)

5. FUTURE SCOPE

This research was performed on a static dataset, it can be performed on dynamic datasets as well. For this purpose, we need a huge amount of data. Further research can be carried out under deep learning and be implemented for real-time and cost-effective fraud detection systems. As this data is very old, new datasets can be used for the same purpose and check the accuracy of different models to get a better result.

6. CONCLUSION

In this project, we applied machine learning algorithms to predict genuine and fraudulent transactions. For doing so, we collected data from the Kaggle website which had around 284,807 transactions out of which only 492 were fraudulent. We learned that the data is highly imbalanced and that must be balanced by using sampling methods. Our main aim was to compare the best model i.e., Random Forest Classifier which comes under ensemble learning methods and is a bagging method with other three boosting methods that included AdaBoost, CatBoost, and XGBoost Classifiers. The result was clear that Catboost gave us high accuracy in very less time.

REFERENCES

[1] “Credit Card Fraud Detection using Machine Learning Algorithms” by Vaishnavi Nath Dornadulaa , Geetha S published by INTERNATIONAL CONFERENCE ON RECENT TRENDS IN ADVANCED COMPUTING 2019, ICRTAC 2019

[2] "Credit Card Fraud Detection Using Machine Learning And Data Science" by Bhargava R1 , Ajay Kumar K2 , Bhavana R3 , Sai Charan S4 , S Lokeswara5

[3] "Machine Learning Model for Credit Card Fraud Detection- A Comparative Analysis" by Pratyush Sharma, Souradeep Banerjee, Devyanshi Tiwari, and Jagdish Chandra Patni School of Computer Science, University of Petroleum and Energy Studies Dehradun published by The International Arab Journal of Information Technology, Vol. 18, No. 6, November 2021

[4] "Credit Card Fraud Detection using Machine Learning and Data Science" by Aditya Saini, Swarna Deep Sarkar Shadab Ahmed Department of Computer Science and Engineering SRM Institute of Science and Technology published by International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Vol. 8 Issue 09, September-2019

[5] "Review on Credit Card Fraud Detection using Machine Learning Algorithms" by Pooja, Dr. Ashlesha-J.C. Bose University of Science and Technology YMCA at Faridabad, Haryana - India published by International Journal of Computer Trends and Technology (IJCTT) – Volume 68 Issue 6 – June 2020