INTERNATIONAL JOURNAL OF TECHO-ENGINEERING

ISSN : 0975-4520

CREDIT CARD FRAUD DETECTION USING XGBOOST CLASSIFIER

O. RAJU

Assistant Professor, Dept. of CSE, JNTUH College of engineering, Jagtial,

oggularaju.jntuh@gmail.com

Abstract: *Credit Card Fraud detection is a challenging task for researchers as fraudsters are innovative, quick-moving individuals. The credit card fraud detection system is challenging as the dataset provided for fraud detection is very imbalanced. In today's economy, credit card (CC) plays a major role. It is an inevitable part of a household, business & global business. While using CCs can offer huge advantages if used cautiously and safely, significant credit & financial damage can be incurred by fraudulent activity. Several methods to deal with the rising credit card fraud (CCF) have been suggested. In this paper, an ensemble learning-based an intelligent approach for detecting fraud in credit card transactions using XGboost classifier is used to detect credit card fraud, and it is a more regularized form of Gradient Boosting. XGBoost uses advanced regularization (L1 & L2), which increases model simplification abilities. Furthermore, XGBoost has an inherent ability to handle missing values. When XGBoost encounters node at lost value, it tries to split left & right hands & learn all ways to the highest loss. The experiments are conducted on the real-time publicly available kaggle dataset with 284,807 credit card transactions included 8 and 31 columns. The experimental results show that the proposed scheme provides better accuracy compared with the previous algorithms.*

Keywords: *Credit card fraud detection, XGboost classifier, fraud detection, machine learning class imbalance.*

I. INTRODUCTION

Credit card fraud is a huge pain and comes with huge fees for banks and card provider companies. Financial companies try to prevent account abuse by using individual security responses. The more complex the security responses, the more fraudsters applying to obtain scammers, i.e. Change their strategies over time. Therefore, it is necessary to improve fraud detection strategies along with security units trying to prevent fraud. Fraud detection has become an essential hobby in reducing the

ISSN : 0975-4520

impact of fraudulent transactions on the transfer of services, rates and popularity of the organization. A range of techniques are used to detect fraud, each attempting to block the maximum penalty for the service while keeping false alarm fees to a minimum. Fraud is the price, and detecting it before a transaction is recorded will significantly lower that price, requiring a very accurate device with very few false alarms. Edge and Falcone Sampaio [1] point out that while implementing proactive methods will increase the likelihood of early warning of fraud, real-time processing dramatically reduces the available time window during which computational analysis must be fashionable and correct selection must be made. As a reaction to the transactions of new arrivals. The faster a fraud detector responds, the better. Fraud detection systems are trained to use old transactions that allow you to decide on new ones. This section of education is a waste of time and can be paralleled in extreme cases. To reduce the computation time, you can still reduce a variety of previous transactions processed with the help of reducing the time window, using less complex methods, etc. Each can lead to lower accuracy, which means more fraud cases and higher false alarms. Therefore, there is a need for a practical device through which the fraud detection system wants to execute and manage transactions in the least possible time. Various fraud detection strategies have been used with the Bayes algorithm [2], neural network [3], Markov model, account signature, and artificial immune systems (AIS). The AIS is mainly based on the human immune system and is similar to a fraud detection device in many elements.

The credit card is a scam, while some other users use your credit card without your permission. Scammers steal your credit card PIN or account information to make any unauthorized transactions without stealing the original physical card. Using credit card fraud detection, we were able to find out if new transactions were fraudulent or genuine.

Number of internet users in India from 2015 to 2020 with a forecast until 2025:

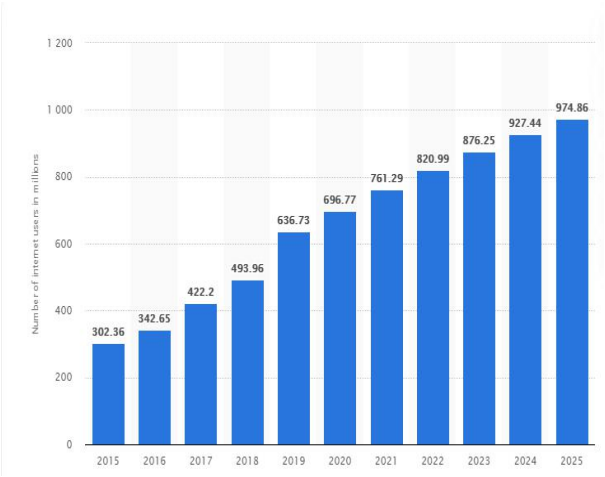


Fig.1 **Number of internet users in India**

ISSN : 0975-4520

In credit card transactions, we've built a device that stumbles upon fraud. The machine has the maximum number of essential characteristics necessary to recognize fraudulent and legitimate transactions. When technical modifications are made, it will be difficult to track the behaviour and presentation of fraudulent transactions. We just noticed the fraudulent activity, but we couldn't prevent it. It is not always easy in real-time to stop known and unknown fraud, but it is possible. The proposed structure is primarily designed to default to online payment credit card fraud, emphasizing moving a fraud prevention device to verify the transaction as fraudulent or legitimate. It is believed that the company and the acquiring bank are related to each other differently for the executive functions. To run this software in real-time, sharing good practices and increasing customer focus among humans can go a long way in reducing losses due to fraudulent transactions.

II. TYPES OF CREDIT CARD FRAUDS

Credit card fraud is a variety of fraud or illegal actions to credit card payment in an automatic payment method. It is unauthorized use of card data or card without the owner's consent [4].

As shown in Figure 1, there are a many methods to commit credit card fraud, namely [5],

1) Identity theft, which is the most common one, is done though using someone's personal information or by entering the existing account.

2) False Cards, also called fake cards, are developed by skimming the actual data from genuine card that has been swiped on an EDC machine.

3) Stolen/Lost Cards are also misused if found by dishonest people or even criminals.

4) Fraud CNP is a type of fraud where the criminal requires minimal information such as card number and expiry date.

5) Clean Fraud is when the purchases are made with stolen cards and later the transactions are changed finding a way around the SDS.

6) Friendly Fraud is when the actual cardholder makes the purchases, pays for them, then files a complaint indicating the loss of the card and claims the refund.

7) Affiliate Fraud is done through making purchases using a fake account or a program that are designed to conduct fraud activities.

ISSN : 0975-4520

8) Triangular fraud which involves three main steps; creating a fake website, providing real or fake offers then using stolen or counterfeit cards to make payments.

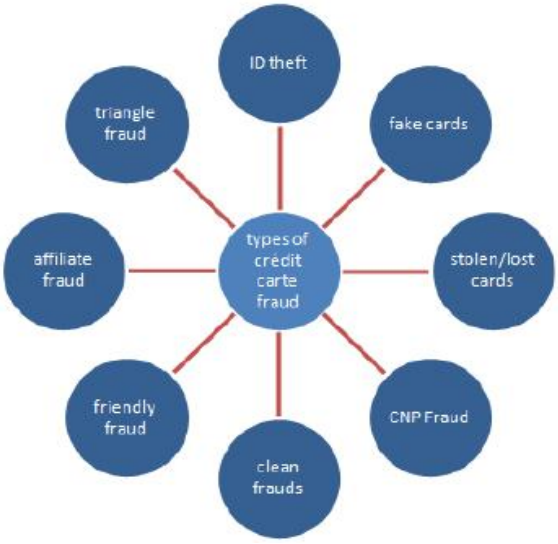


Fig.2 Types of credit card frauds

The figure .2 shows the types of credit card frauds

III. PROPOSED METHODOLOGY

First the credit card dataset is taken from the supply, and cleaning and approval is executed on the dataset which joins disposal of excess, filling void territories in sections, changing imperative variable into components or exercises then actualities is part into 2 sections, one is preparing dataset and another is check data set. Presently k crease move approval is done that is the special example is arbitrarily divided into k same and equivalent measured subsamples.

XGBoost Algorithm

XGBoost has been widely used in many fields to achieve state-of-the-art results on some data challenges (e.g., Kaggle competitions), which is a high effective scalable machine learning system for tree boosting. XGBoost is optimized under the Gradient Boosting framework and developed by Chen and Guestrin [18], which is designed to be highly efficient, flexible and portable. The main idea of boosting is to combine a series of weak classifiers with low accuracy to build a strong classifier with better classification performance. If the weak learner for each step is based on the gradient direction of the loss function, it can be called the Gradient Boosting Machines.

XGBoost is an efficient and scalable implementation of the Gradient Boosting Machine (GBM), which has been a competitive tool among artificial intelligence methods due to its features such as easy parallelism and high prediction accuracy. Furthermore, the following advantages make it adaptable to deal with the transient stability prediction:

(1) In XGBoost model, multithreading parallel computing can be automatically called, which is faster than the traditional ensemble learning to predict the transient

ISSN : 0975-4520

stability with large amounts of data in the actual power grid.

(2) That the regularization term addition to XGBoost, makes its generalization ability be improved, which makes up for the shortcoming that the decision tree is easy to be over-fitted. (3) XGBoost is the tree structure model, which doesn't need to normalize the data collected by PMU in the power system. Furthermore, it can effectively deal with the missing values, which is suitable for PMU-based transient stability prediction to discover the relationship between features and transient stability.

SYSTEM ARCHITECTURE

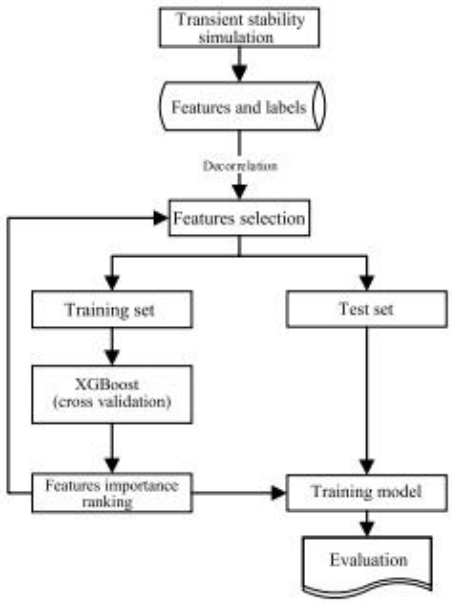


Fig.3. Credit card fraud evaluation process of proposed work

As shown in the figure.3, it shows the credit card fraud evaluation process for the proposed work using the XGboost classifier. The below algorithm provides a brief classifier process of the proposed XGboost algorithm.

Algorithm 1: XGboost classifier algorithm

Input: Dataset $\mathbf{D} = \{(x_i,y_i):i = 1....n,\ x_i \in \Re^m, y_i \in \Re\}$

we have n samples with m features

1. The prediction value model is $\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), \quad f_k \epsilon F$

where f_k is independent regression tree and $f_k(x_i)$ is prediction score given by

k^{th} tree to i^{th} sample

2. The set of functions f_k in the regression tree model can be learned by minimizing objective function:

$Obj = \sum_{i=1}^{n} l(y_i,\hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k)$ where l is training loss function

3.To avoid over − fitting, the term Ω penalizes the complexity of the model:

$$\Omega(f_k) = \gamma T + \frac{1}{2}\lambda||w||^2$$

where γ and λ are the degrees of regularization. T and w are the numbers of leaves

4.Let $\hat{y}_i^{(t)}$ be the prediction of the i^{th} instance at the t^{th} iteration it needs to add f_t to minimize

ISSN : 0975-4520

the following objective:

$$Obj^{(t)} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)\right) + \Omega(f_t)$$

5.To remove the constant term following equ given

$$Obj^{(t)} = \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t(x_i)^2\right] + \Omega(f_t)$$

where $g_i = \partial_{\hat{y}_i^{(t-1)}} l\left(y_i, \hat{y}_i^{(t-1)}\right)$ and $h_i = \partial^2_{\hat{y}_i^{(t-1)}} l\left(y_i, \hat{y}_i^{(t-1)}\right)$ are the first

and the second order gradient on l

6.Then the objective is rewritten as:

$$Obj^{(t)} = \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t(x_i)^2\right] + \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$$

$$= \sum_{j=1}^{T} \left[\left(\sum_{i \epsilon I_j} g_i\right) w_j + \frac{1}{2}\left(\sum_{i \epsilon I_j} h_i + \lambda\right) w_j^2\right] + \gamma T$$

Where $I_j =$ $\{i | q(x_i) = j\}$ denotes the instance set of leaf j,For

tree structure

the optimal weight w_j^* of leaf j

7.The corresponding optimal value can be ca

$$w_j^* = -\frac{G_j}{H_j + \lambda}$$

$$Obj^* = -\frac{1}{2} \sum_{j=1}^{T} \frac{G_j^2}{H_j + \lambda} + \lambda T$$

where $G_j = \sum_{i \epsilon I_j} g_i$, $H_j =$ $\sum_{i \epsilon I_j} h_i$, , obj presents the quality of a tree str

The Matching algorithm (test) is explained below,

Step 1. Count the number of attributes in the incoming transaction matching with that of the legal pattern of the corresponding customer. Let it be lc.

Step 2. Count the number of attributes in the incoming transaction matching with that of the fraud pattern of the corresponding customer. Let it be fc.

Step3.If fc = 0 and lc is more than the user defined matching percen the incoming transaction is legal.

Step4.If lc = 0 and fc is more than the user defined matching percen

the incoming transaction is fraud

Step 5. If both fc and lc are greater than zero and fc ≥ lc, then the incoming transaction is fraud or else it is legal.

The pseudo code of the testing algorithm is given below,

Input:Legal Pattern Database LPD, Fraud Pattern Data T, Number of Customers "n" Number attributes "k" ma

output:**0** (if legal)or **1**(if fraud)

Assumption:

1.First attribute of each record in pattern databases and incoming transaction is Customer ID.

2.if an attribute missing in the frequent item set then we considered it as invalid.

Begin

ISSN : 0975-4520

1. **lc** = 0;//
   legal attribute match count
2. **fc** = 0;//
   fraud attribute match count
3. **for i = 1 to n do**
4. if(LPD(i,1) = T(1)) **then** //
   first attribute
5. **for** j = 2 to k **do**
6. **if**(LPD(i,j) is valid and LPD(i,j) = T(j)) then
7. lc = lc + 1
8. **endif**
9. **endfor**
10. **endif**
11. **endfor**
12. **for i = 1 to n do**
13. if(FPD(i,1) = T(1)) **then** //
    first attribute
14. **for** j = 2 to k **do**
15. **if**(FPD(i,j) is valid and FPD(i,j) = T(j)) then
16. fc = fc + 1
17. **endif**
18. **endfor**
19. **endif**
20. **endfor**
21. **if**(fc = 0) then //
    no fraud pattern
22. if(lc/
    no.of valid attributes in legal pattern mp) **then**
23. return (0); //legal transaction
24. **else** return (1);//
    fraud transaction
25. **endif**
26. **elseif**(lc = 0) then //
    no legal pattern
27. if(fc/
    no.of valid attributes in fraud pattern) ≥ mp) **then**
28. return (1); //fraud transaction
29. **else** return (0);//
    legal transaction
30. **endif**
31. **elseif**(lc > 0&&fc > 0)**then** both legal and fraud pattern available
32. **if**(fc ≥ lc) **then**
33. **return**(**1**);//fraud transaction
34. **else return**(**0**);//
    legal transaction
35. **endif**
36. **endif**
37. **End**

After finding fraud patterns and legal patterns for each customer, the fraud detection system goes through fraud databases and legal fraud detection patterns. These style databases are much smaller than the original customer transaction databases, because they contain only one customer record. This research proposes a matching algorithm that traverses the pattern databases to match the incoming transaction for fraud detection. If the closest match is found to the corresponding customer's legal pattern, the matching algorithm returns "0," this gives a green signal to the bank to allow the

ISSN : 0975-4520

transaction. If the closest match to the customer's fraud pattern is found, the matching algorithm returns "1" which gives the bank an alert to stop the transaction.

IV. DATASET USED

In this section, credit card fraud detection related dataset used from the publicly available kaggle dataset. The dataset contains transactions made by credit cards in September 2019 by European cardholders. This dataset presents transactions that occurred in two days, where we have 492 frauds out of 284,807 transactions. The dataset is highly unbalanced, the positive class (frauds) account for 0.172% of all transactions. The dataset divided into two groups of training set with 70% and testing set with 30%. It contains only numerical input variables which are the result of a PCA transformation. Unfortunately, due to confidentiality issues, we cannot provide the original features and more background information about the data. Features V1, V2, … V28 are the principal components obtained with PCA, the only features which have not been transformed with PCA are 'Time' and 'Amount'. Feature 'Time' contains the seconds elapsed between each transaction and the first transaction in the dataset. The feature 'Amount' is the transaction Amount, this feature can be used for example-dependant cost-sensitive learning. Feature 'Class' is the response variable and it takes value **1** in case of fraud and **0** otherwise.

V. RESULTS AND DISCUSSONS

In this section the experimental conducted on the real-time publicly available kaggle dataset with 284,807 credit card transactions included 8 and 31 columns. In the phase of the experimental stage, we are using Python 3.7 version to perform and evaluate the proposed algorithm.

a) Performance Evaluations:

The proposed diagnostic method is expected in general performance situations using a well-known matrix that includes accuracy, precision, recall, and F1-score. These metrics are calculated using True Positive (TP) , True Negative (TN) , False Positive (FP) , and

False Negative (FN) parameters. When TP is likely to cause most cancers in a cancer patient, FPs will likely find that the rate at which cancer is detected is the rate at which a healthy person is found. TN hopes to reveal that a person with cancer is healthy. FN is prescribed when a healthy man or woman has cancer.

ISSN : 0975-4520

b) Evaluation Metrics used:

Accuracy: Accuracy is a measure of the overall effectiveness of a rating system. The following equation may be considered.

$$\text{Accuracy} = \frac{\text{True Positive+True Negativee}}{\text{True Positive+False Positive+True Negative+False Negative}} 100$$

Precision: The precision scale shows the expected number of nodules associated with cancer.

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive+False Positive}}$$

Recall: Recall is called the ability to classify exquisite styles. The following equations can be used to harvest this.

$$\text{Sensitivity} = \frac{\text{True Positive}}{\text{True Positive+False Positive}}$$

F1-score: The F1-score, also known as the F1-measure, it is a measure of a model's accuracy on a dataset. The below equation used to calculate f1-score.

$$2 * \frac{\text{Precision}*\text{Recall}}{\text{Precision+Recall}}$$

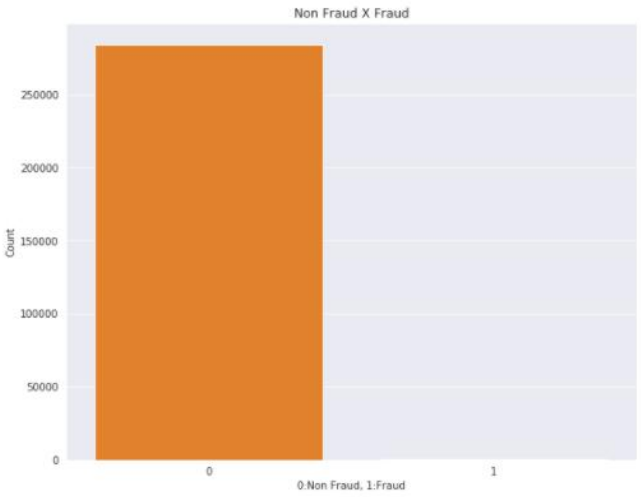
c) EXPERIMENTAL RESULT:

Table.1 Total rows and columns in table

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	V24	
count	284807.0	284807.0	284807.0	284807.0	284807.0	284807.0	284807.0	284807.0	284807.0	284807.0	...	284807.0	284807.0	284807.0	284807.0	2848
mean	94814.0	0.0	0.0	-0.0	0.0	-0.0	0.0	-0.0	-0.0	-0.0	...	0.0	0.0	0.0	0.0	
std	47488.0	2.0	2.0	2.0	1.0	1.0	1.0	1.0	1.0	1.0	...	1.0	1.0	1.0	1.0	
min	0.0	-56.0	-73.0	-48.0	-6.0	-114.0	-26.0	-44.0	-73.0	-13.0	...	-35.0	-11.0	-45.0	-3.0	-1
25%	54202.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-0.0	-1.0	...	-0.0	-1.0	-0.0	-0.0	
50%	84692.0	0.0	0.0	0.0	-0.0	-0.0	-0.0	0.0	0.0	-0.0	...	-0.0	0.0	-0.0	0.0	
75%	139320.0	1.0	1.0	1.0	1.0	1.0	0.0	1.0	0.0	1.0	...	0.0	1.0	0.0	0.0	
max	172792.0	2.0	22.0	9.0	17.0	35.0	73.0	121.0	20.0	16.0	...	27.0	11.0	23.0	5.0	

8 rows × 31 columns

Table.1 indicates the credit card transactions details with 8 rows and 31 columns. Features V1, V2, … V28 are the principal components obtained with PCA, the only features which have not been transformed with PCA are 'Time' and



'Amount'. Feature 'Time' contains the seconds elapsed between each transaction and the first transaction in the dataset.

```
Non Fraud %  99.83

count    284315.00
mean         88.29
std         250.11
min           0.00
25%           5.65
50%          22.00
75%          77.05
max       25691.16
Name: Amount, dtype: float64
```

Fig. 4 Non fraud transactions in %

ISSN : 0975-4520

```
Fraud %     0.17

count     492.00
mean      122.21
std       256.68
min         0.00
25%         1.00
50%         9.25
75%       105.89
max      2125.87
Name: Amount, dtype: float64
```

Fig.5 Fraud transactions in %

As shown in the figure 4 and 5, the fraud and non- fraud transaction are found using feature selection process using proposed method. The non-fraud transaction are found with 99.83% (Fig.4) and fraud transactions found with 0.17% (Fig.5).

Fig.6 Graphical representation of fraud and non-fraud transactions

The figure.6 shows the graphical representation of fraud and non-fraud transactions found from given database. The X-axis indicates the count of transactions and Y-axis indicates the non-fraud and fraud transactions. As shown in the figure, the value 0 indicates the non-fraud transactions and 1 indicates the fraud transactions.

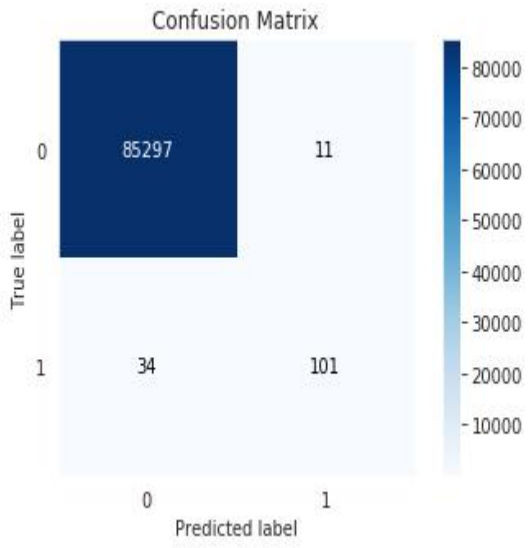


Fig.7 Confusion matrix

As shown in the figure.7, a confusion matrix is an N x N matrix used for evaluating the performance of a classification model. The matrix compared the true label values with those predicted by the XGboost model.

	precision	recall	f1-score	support
0	1.00	1.00	1.00	85308
1	0.90	0.75	0.82	135
accuracy			1.00	85443
macro avg	0.95	0.87	0.91	85443
weighted avg	1.00	1.00	1.00	85443

Fig. 8 Accuracy for the proposed model

The figure.8 shows the proposed model performance metrics using precision is

ISSN : 0975-4520

90%, recall is 75%, f1-score is 82%, and accuracy with 100%.

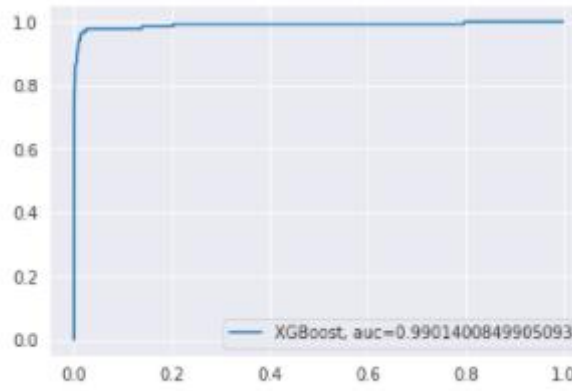


Fig.9 XGboost Classifier final result

As shown in the fig.9, the classifier had a very good result, with AUC of 0.99.

Table.2 Performance comparison between various classifiers

Author with Year	**Model**	**Accuracy**
Altyeb et al.[2020]	(OLightGBM)	98.40%
Lakshmi et al.[2018]	RF	95.5%
Proposed method	XGboost classifier	99%

Table 2 indicates the comparison between the various classifiers such as optimized light gradient boosting machine (OLightGBM), Random forest (RF), and proposed XGboost classifier.

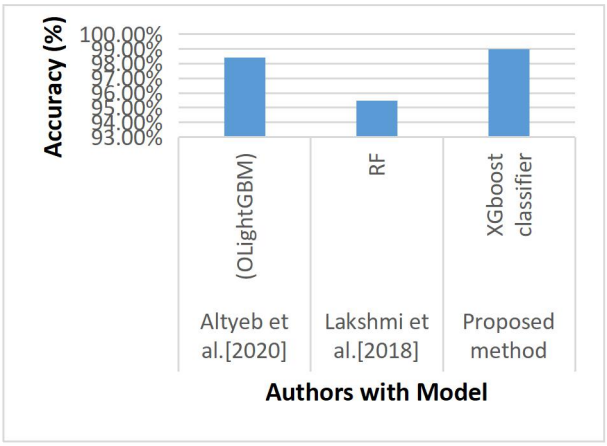


Fig.10 Accuracy comparison between various models

As shown in figure.10, the performance of the proposed model comparison is taken between various traditional algorithms such as optimized light gradient boosting machine (OLightGBM), Random forest (RF).

VI. CONCLUSION

We reached a very satisfactory number in detecting fraud transactions in relation to the initial model, rising from 75% to 98% of correctly identified transactions. In return, the detection of correctly identified normal transactions decreased from 99% to 97. Remember that we need to determine where this exchange is worthwhile. Generally, the costs of losing a fraudulent transaction are often greater

ISSN : 0975-4520

than mistakenly classifying a good transaction as fraud. One of the challenges is to find the balance in training your model and proceed accordingly. It is evident from the findings described in the paper that XGBoost works well in both static and incremental installations. We produce realistic synthetic data that are working on our research project because abundant data sets are not publicly available. The results thus obtained showed that the highest precision and accuracy of XGBoost is 99.01% when compared with the traditional models of optimized light gradient boosting machine (OLightGBM), Random forest (RF).

REFERENCES

[1] M.E. Edge, P.R. Falcone Sampaio, A survey of signature based methods for finan-cial fraud detection, Comput. Secur. 28 (2009) 381–394.

[2] S. Panigrahi, A. Kundu, S. Sural, A. Majumdar, Credit card fraud detection: a fusion approach using Dempster–Shafer theory and Bayesian learning, Inf.Fusion 10 (2009) 354–363.

[3] A. Krenker, M. Volk, U. Sedlar, J. Beˇster, A. Kos, Bidirectional artificial neural networks for mobile-phone fraud detection, ETRI J. 31 (2009) 92–94.

[4] A. Khan, N. Akhtar, and M. Qureshi, “Real-Time Credit-Card Fraud Detection using Artificial Neural Network Tuned by Simu-lated Annealing Algorithm,” nt. Conf. Recent Trends …, 2014

[5] Rajeshwari U and B. S. Babu, “Real-time credit card fraud detection using Streaming Analytics,” 2016 2nd Int. Conf. Appl. Theor. Comput. Commun. Technol., pp. 439–444, 2016.

[6] Roy, Abhimanyu, et al, 2018, “Deep learning detecting fraud in credit card transactions”, Systems and Informat ion Engineering Design Symposium (SIEDS), IEEE.

[7] Andrew Dahbura, Stephen Adams, 2019, "Improving Credit Card Fraud Detection by Profiling and Clustering Accounts", Systems and Information Engineering Design Symposium (SIEDS), pp. 1-6.

[8] Akhil Sethia, Purva Raut, 2018, "Data Augmentation using Generative models for Credit Card Fraud Detection", Computing Communication and Automation (ICCCA) 2018 4th International Conference on, pp. 1-6.

[9] Sangeeta Mittal, Shivani Tyagi, "Performance Evaluation of Machine Learning Algorithms for Credit Card Fraud Detection", Cloud Computing Data

ISSN : 0975-4520

Science & Engineering (Confluence) 2019 9th International Conference on, pp. 320-324, 2019.

[10] R. S. Miguel Carrasco, Miguel-Ángel Sicilia-Urbán, 2020, "Evaluation of Deep Neural Networks for Reduction of Credit Card Fraud Alerts", Access IEEE, vol. 8, pp. 186421-186432.

[11] C. Rohini Sri Vasavi, Kothuri Praharshitha, 2021, "Survey on Detection of Credit Card Frauds using Hmm and various Clustering Approaches", Inventive Computation Technologies (ICICT) 2021 6th International Conference on, pp. 101-107, 2021.