Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.776522

COMPARING VARIOUS CREDIT CARD FRAUD DETECTION METHODS BASED ON MACHINE LEARNING ALGORITHMS

Neha Purohit and Dr. Rajeev G. Vishwakarma
Department of Computer Science & Engineering, Dr. A.P.J. Abdul Kalam University, Indore (M.P.) 452010, India
Corresponding Author Email : nehapurohit059@gmail.com

Abstract— Credit card fraud detection is one of the essential tasks of the banking system. However, there are various Machine Learning (ML) solutions available for detecting and preventing potential fraud. In this paper, the aim is to compare the recently contributed ML-based credit card fraud detection technique. In this context, three different models are considered, first a brief introduction of the considered approaches has been given. Then, based on experimental results the comparison among the techniques has been discussed. Based on the findings, the publically available credit card fraud detection dataset has suffered from the class imbalance problem, large dimensions, missing values, overlapped attributes, and outliers. Therefore, two key approaches are discussed first is based on handling the class imbalance problem and the second is based on deep cleaning of the dataset. According to the results, we found that the deep clean method is superior then the over-sampling technique. The deep clean technique reduces the dimensions and also removes the noise from the dataset. Therefore the method reduces the data processing cost in terms of time and memory. Additionally, the classification accuracy has a similar behavior as the oversampling. In both cases, the performance varies between 97.9%- 99.2%. Finally, based on the comparative study the conclusion of the work has been made.
Keywords— Machine Learning, Classification, Credit card fraud detection, machine learning application, supervised and unsupervised learning, comparison.

I. INTRODUCTION

The Indian digital payment systems are become much popular in recent years. This system is being adopted by the common person during demonetization and Covid-19 crisis. This rising online payments are also attracting the hackers and attackers to conduct frauds. Among the different digital payment methods the credit card is one of the leading payment options in India. Therefore credit card fraud detection is one of the issues in banking companies [1]. In India most of the fraud cases are performed due to less awareness of digital channels [2]. In literature a number of ML based credit card fraud detection techniques are available. The aim of these techniques to capture the patterns of credit card usages and estimate the possible fraud cases [3].

In this paper, we discuss the recently contributed ML based credit card detection techniques. With aim to accomplish the following objectives:

(A) **Implementation of ML algorithms on credit card fraud detection Dataset:** In this phase, the seven ML algorithms are considered and applied on credit card fraud detection dataset. Additionally the issue in dataset analysis has been recovered.

(B) **Implementation of Deep learning algorithms on credit card fraud detection Dataset:** In this phase, the problem of class imbalance is considered. Therefore, the data over sampling has been performed and the Convolutional Neural Network (CNN) has been applied for training and validation. During this study we have found the data has a higher dimension and also contains the noise.

(C) **To develop and design novel algorithm for credit Card Fraud Detection:** In this phase, an algorithm is proposed for performing deep leaning operation on dataset. Additionally, two classification techniques have been considered namely CNN and XgBoost.

(D) **To compare this algorithm and its results with existing ML algorithms:** In this phase the comparative performance study has been conducted to find an appropriate technique for handling the credit card fraud detection technique. The aim is to find an efficient and accurate technique for credit card fraud detection.

In this paper, this section provides the overview of the presented work. Next section provides the discussion about the recently introduced work based on ML and deep learning. Finally the experiments are carried out and the comparative study has been conducted. Based on the experimental analysis the conclusion has made.

II. PROPOSED WORK

This section involves the brief description of the three different ML based models for credit card fraud detection. Additionally their functional aspects are also described.

1. A simple ML based Fraud Detection

The simple ML based credit card fraud detection technique is given in figure 1. The details of their components are given as:

Input Training samples: The credit card issuer company is responsible to recognize fraudulent transactions, so customers are not charged for any such fraudulent activity. The training dataset consist transactions done on Sep 2013 by European cardholders. It has a total of 284,807 transactions among 492 frauds transactions are also given. It contains numerical values obtained by Principal Component Analysis (PCA), due to security the original features are hidden. Only features 'Time' and 'Amount' is kept as it is. Feature 'Class' is predictable variable and has 1 for fraud and 0 for normal transactions.

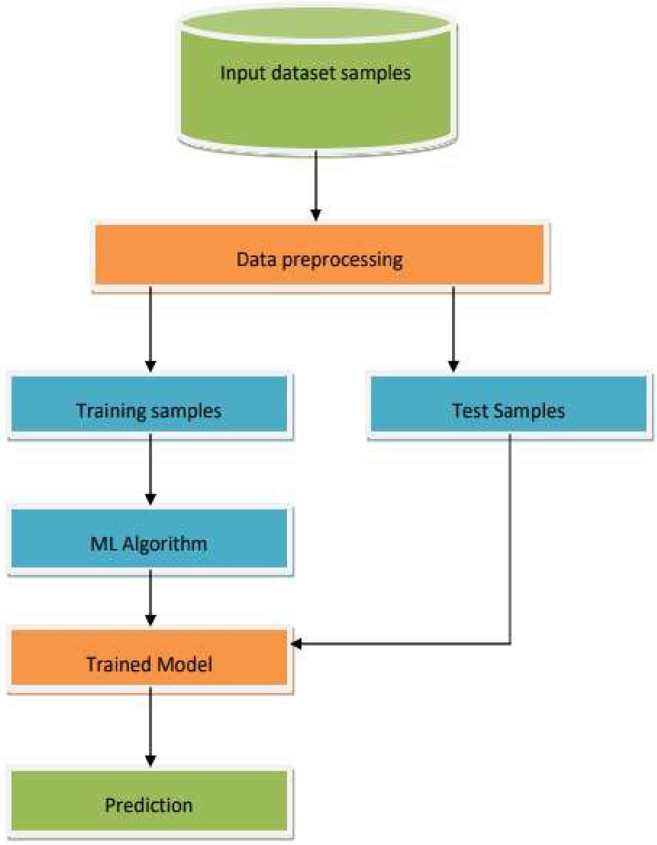

Figure 1: Proposed model

Data pre-processing: In data pre-processing different techniques are applied to clean the data. In this work, first we visualized the data. Figure 2 demonstrate the initial plot of the data samples. In order to pre-process the dataset we have used correlation coefficient among the dataset features and the class labels of the dataset.

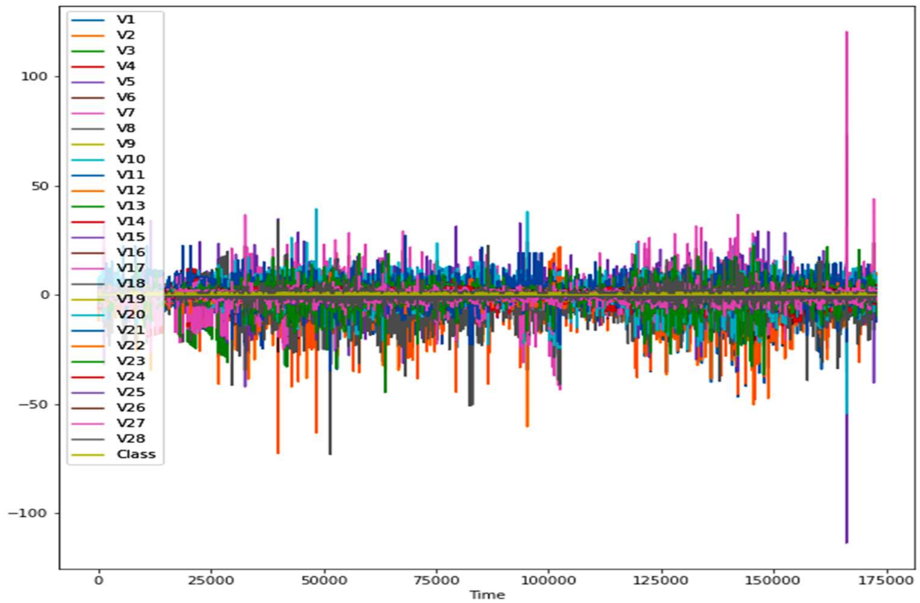

Figure 2: Initial Data Samples

The correlation coefficient can be described by:

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}} \dots \dots \dots \dots (1)$$

Where, x_i and y_i are the sample values, $\bar{x}$ and $\bar{y}$ are the mean.

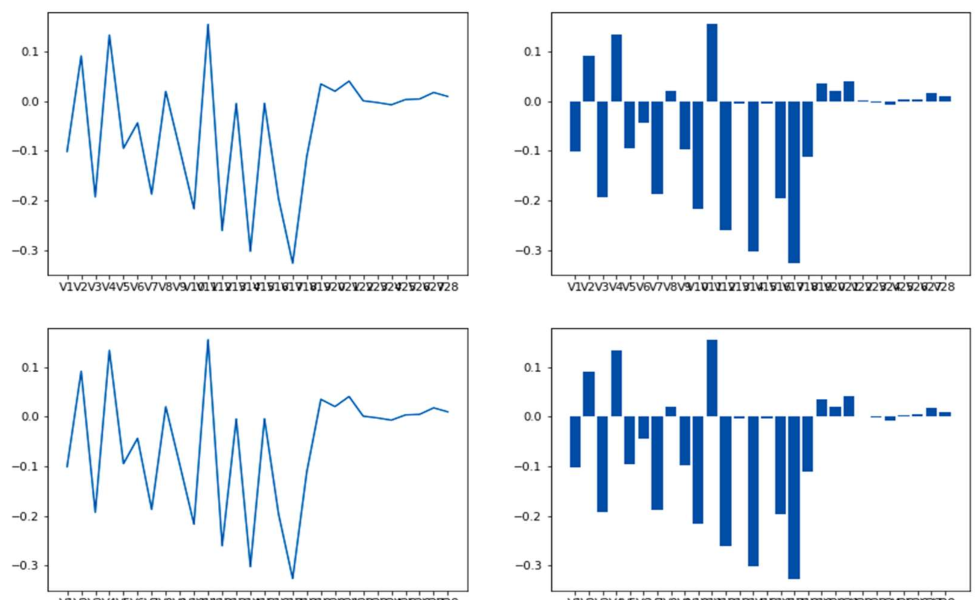

Figure 3: correlation coefficient for dataset attributes

The correlation coefficient of the different attributes is given in figure 3. Based on correlation coefficient we have filtered the attributes which are having less relevant information. After data pre-processing only 22 attributes and one class label is remain. Next, dataset is split into two parts i.e. training and testing. The training set contains 75% of samples and 25% of samples are used for validation. Finally, the training samples are used to train the ML models. The

Logistic Regression (LR), Random Forest (RF), K nearest Neighbors (KNN), Gaussian naive Bayes (GNB), Support vector machine (SVM), K-means, and Convolutional neural network (CNN) is used for training. After training the validation set is used for classification and result analysis.

2. Handling Class Imbalance problem

The same dataset is used as the previous phase. In this dataset, due to PCA-based transformation, the data has "+" positive and negative "-" values. Therefore, we need to scale the dataset values between 0-1. For this task min-max normalization is used. The dataset has a total of 30 attributes. Therefore, we have performed chi-square test between class and other attributes. The chi-square test returns chi-square score and p values.

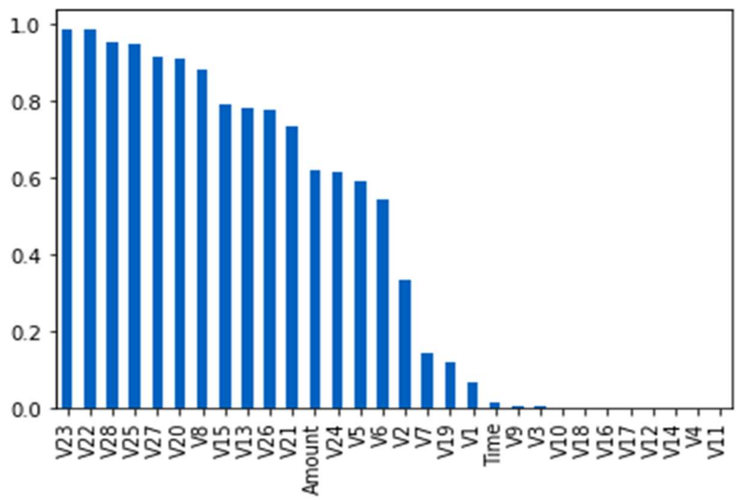

Figure 4: p value based attribute ranking

The sorted p values-based attribute ranking is given in figure 4. According to p values attributes 'V11', 'V4', 'V14', 'V12', 'V17', 'V16', 'V18', and 'V10' has less significant. Therefore, we eliminate them, and remaining 22 attributes are used. Next, dataset is checked for class imbalance. During this, we found 99.83% of samples are belongs to class '0', and only 0.17% of samples are class '1'. Thus data is highly imbalance. Figure 5 demonstrates the majority and minority classes.

Figure 5: Class distribution of dataset

Therefore, we perform balancing of the classes. In this context, the classes are balanced using the sampling technique. There are two types of techniques available for class balancing i.e. over sampling and under sampling. The under sampling technique has the risk of information loss, therefore we are utilizing over sampling technique. Thus, Adaptive Synthetic Sampling (ADASYN) and Synthetic Minority Oversampling Technique (SMOTE) are used. After applying the ADASYN total samples become 564546 instances and further divided into 75% of training samples (426409) and 25% of testing samples (142137). Additionally, using the SMOTE total samples becomes 568630 instances. Additionally the training samples 75% becomes 426472 instances and 25% of samples are becomes 142158 instances. The class distribution generated by both the methods is given in figure 6.

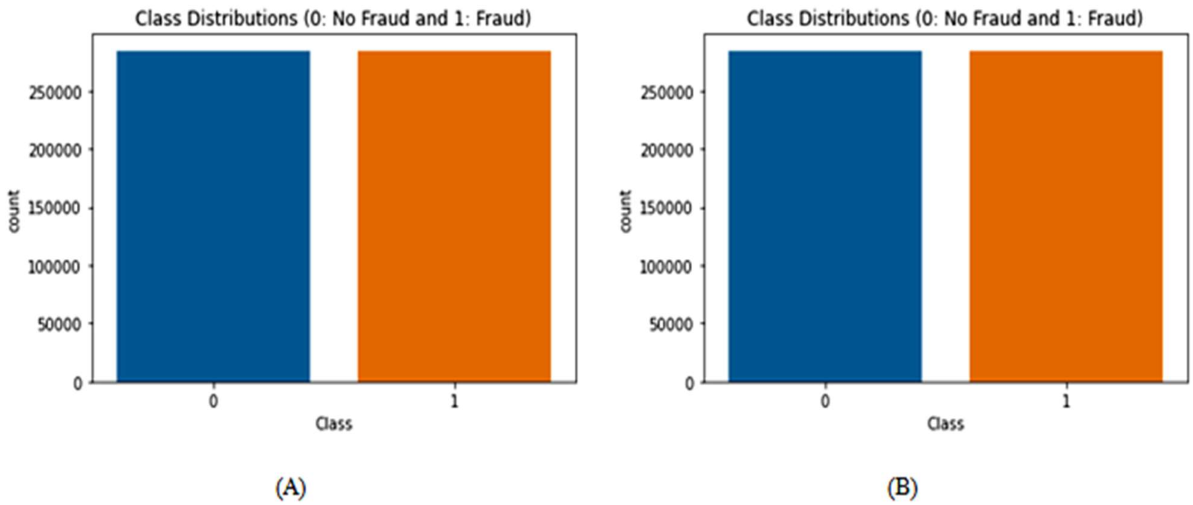

Figure 6: shows the samples generated by (A) ADASYN and (B) SMOTE

Next, we configured a sequential Convolutional neural network (CNN). This network has an Input Layer: type dense, number of neurons = 100, input dimensions = 22, activation function = ReLu. Next five hidden layers are used with type dense, and activation function sigmoid and ReLu is used alternately. Finally the output layer is used with type dense, number of neurons = 2, activation function = SoftMax. Additionally, to compile the network Loss Function = 'categorical_crossentropy', Optimizer = 'adam' and Metrics = 'accuracy' is used. The same neural network is used for both kinds of samples prepared by SMOTE and ADASYN.

3. Handling outlier in dataset

The same dataset is used as the previous two experiments. Next, we performed chi-square test. According to the p values the attributes 'V23', 'V22', 'V28', 'V25', and 'Amount' has been selected for further experiment. Additionally, we have removed the less significant attributes. Let the remaining credit card data set is $D = \{D_1, D_2, \dots, D_n\}$ with class a C. But the dataset may contain missing value therefore a new data set D_a is generated by replacing the frequent value. Table 1 shows the process to handle missing values. The dataset D_a is first read to get the dimension of data in terms of the total number of rows and columns. Now for each attribute we calculate the most frequent value. Using frequently identified values we replace the NaN value. After replacing all the missing values we found a new dataset D_c. But, the data has "+" positive as well as negative "-" values. Therefore, we utilized min-max normalization. The steps used for normalizing the dataset are also given in table 1. According to the given process first maximum and minimum values are extracted from the dataset. Additionally by using the min-max normalization technique the dataset values are normalized for creating a new dataset D_{norm}. Now in order to increase separability of dataset we need to find the overlapped attributes.

The overlapped attributes may negatively impact on classification accuracy. In this context, the Mega Trend Diffusion (MTD) is used to analyse the dataset. The MTD is a fuzzy based technique for finding the samples which are overlapped. In order to explain this concept let the attribute $A = \{A_1, A_2, \ldots, A_n\}$, with two boundary conditions "a" and "b". The approximation of these conditions is performed using equation (2) and (3):

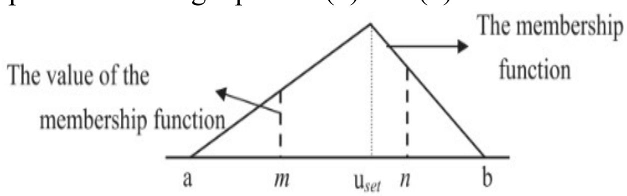

Figure 7: MTD function

$$a = u_{set} - skew_L * \sqrt{(-2) * \frac{s_a^2}{N_L * \ln(f(t))}} \ldots \ldots (2)$$

$$b = u_{set} - skew_U * \sqrt{(-2) * \frac{s_a^2}{N_U * \ln(f(t))}} \ldots \ldots (3)$$

Table 1: Missing Value Handling

Input: Dataset D_a **Output:** clean Dataset D_c

Process:
 	
1. $[col, row] = ReadDataset(D_a)$
2. $for(i = 1; i \leq col; i++)$
    a. $f = GetFrequent(D_a[i])$
    b. $for(j = 1; j \leq row; j++)$
        i. $if\ D_a[i][j] == null \mid\mid D_a[i][j] = NaN$
            1. $D_a[i][j] = f$
        ii. End if
    c. End for
    d. $D_c.Append(D_a[i])$
3. End for
4. $for(i = 1; i \leq col; i++)$
    a. $max = findMax(D_c[i])$
    b. $min = findMin(D_c[i])$
    c. $for(j = 1; j \leq row; j++)$
        i. $val = D_c[i][j]$

ii. $newVal = \frac{val - min}{max - min}$
iii. $D_{norm}[i][j] = newVal$
d. End for
5. End for
6. Return D_{norm}

Where,

$$u_{set} = \frac{min + max}{2} \dots \dots ..(4)$$

$s_a^2 =$ Variance of attribute A_i
N_L= the number of data points smaller than u_{set}
N_U= the number of data points greater than u_{set}

$$skew_L = \frac{N_L}{N_L + N_U} \dots \dots ..(5)$$

and,

$$skew_U = \frac{N_U}{N_L + N_U} \dots \dots ..(6)$$

And, $f(t)$ is a real number greater than 0.
Next we define the MTD as membership function, which is denoted by $m(x)$ as given in equation (7).

$$m(x) = \begin{cases} \frac{x-a}{u_{set}-a}, & a \le x \le u_{set} \\ \frac{b-x}{u_{set}-b}, & u_{set} \le x \le b \\ 0, & otherwise \end{cases} \dots \dots ..(7)$$

Now, we need to calculate the overlap area to and deciding the high or low overlapped area. In our experimental dataset credit card fraud detection we have two classes F for fraud and T for legitimate. The area of MDT function of attribute A_i is β_A^i and for the same A_i for class B is given by β_B^i. Then the overlapped area of class F and T is β_O^i. Thus the rate of overlap of class F is given by β_O^i/β_A^i and for class T is β_O^i/β_B^i. Then degree of overlap is calculated by equation (8).

$$OD^i = \sqrt{\frac{\beta_O^i}{\beta_A^i} * \frac{\beta_O^i}{\beta_B^i}} \dots \dots ..(8)$$

Then a threshold for OD^i is calculated as the mean of OD=($OD^1, \dots, OD^i, \dots OD^n$). The corresponding attributes are defined as having low overlap when less than threshold T:

$$T = \frac{1}{n}\sum_{i=1}^{n} OD^1 + OD^i + OD^n \dots \dots ..(9)$$

The high and low overlap area is defined by using.

$$\begin{cases} OD^i < T, & LO \\ OD^i > T, & hO \end{cases} \dots \dots ..(10)$$

The overlapping condition shows the quality of data attributes. However, MTD involves two processes to deal with high and low overlap data. In this context, only low overlapped attributes 'V23', 'V22', 'V28', 'V25', and 'Amount' has been selected. Further for improving the data quality the outlier analysis performed. The outlier is the data points that are providing misleading patters. Basically, it is a spike or downfall in trend. Therefore, the data is used with the regression analysis. The regression analysis is used to measure:

$$[R, R_i] = regress(D) \dots \dots (11)$$

Where, R is residual of size n-by-1, R_i is n-by-2 matrix of intervals used to diagnose outliers. If interval R_i (i, :) for observation i does not include zero, therefore residual is larger than expected is suggested as outlier. The outlier is an error in the data set. In order to eliminate the outliers we follow steps given in table 2. The given algorithm accept the dataset D_{red} and generate the outlier free data P_{out}. Each instance is verified using the residual values. If residual both the intervals are below or higher than zero then the data instance is considered as outlier.

Table 2 outlier detection

Input: reduced dimension of data D_{red} **Output:** outlier points P_{out}

Process:
1. $[col\ row] = Dataread(D_{red})$
2. $[R, R_i] = regress(D_{red})$
3. $for(i = 1; i \leq row; i++)$
    a. $if(R_{i,1} \leq 0\ and\ R_{i,2} \leq 0)$
        i. $P_{out}.Add(D_{red}[i])$
    b. $else\ if\ (R_{i,1} \geq 0\ and\ R_{i,2} \geq 0)$
        i. $P_{out}.Add(D_{red}[i])$
    c. End if
4. End for
5. Return P_{out}

After preparing the final data we utilize two ML algorithms Xgboost and CNN to train and classify data. This section provides the details about the prepared credit card fraud detection techniques. The next section discusses the experimental results of the systems.

III. RESULTS & DISCUSSION

The aim of this study is to identify the appropriate technique for designing the credit card fraud detection. However, there are limited experimental dataset is available additionally have the different issues which make the classification task complex. These issues are:

1. class imbalance issue
2. outlier
3. attribute overlapping issues

Therefore, the three experimental scenarios can be considered in this study:

1. Experiment with actual dataset

In order to perform the experiments there are two key parameters are considered namely accuracy and training time. The accuracy is the ratio of correctly recognized information and total information for recognition. The accuracy can be measured using:

$$accuracy = \frac{correctly\ recognized}{total\ samples} \ldots\ldots\ldots\ldots.(12)$$

Additionally, the training time is given as the amount of time taken by the algorithm for performing the experiments. That can be calculated using:

$$Training\ time = training\ end\ time - start\ time \ldots..(13)$$

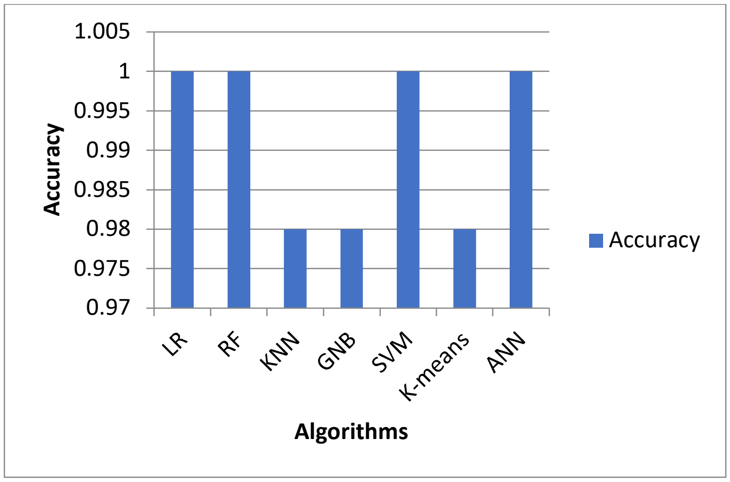

Figure 7: Accuracy with actual dataset

According to the obtained accuracy of the ML algorithms as given in figure 7, we can see the algorithms are providing the higher accuracy. Here the KNN, GNB and k-means are providing 98% accuracy and LR, RF, ANN and SVM are providing the accuracy 100%. But the dataset is highly imbalance thus the accuracy is not a suitable parameter for performance analysis. The next parameter is training time which is demonstrated in figure 8. According to the obtained training time the KNN, GNB and k-means are consuming the less amount of time as compared to other algorithms. Additionally the SVM and RF are highly time consuming algorithms for performing the training. Therefore, in order to verify the actual performance we need to rectify the data issue before make use for developing the required fraud detection system.

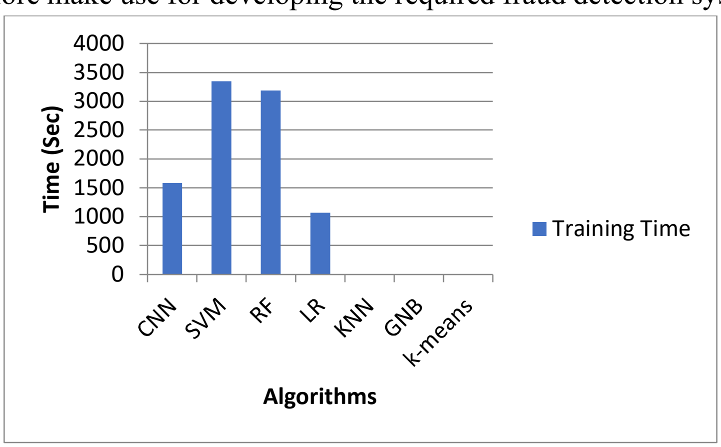

Figure 8: training time with the actual dataset

2. Experiment after balancing dataset

In this experiment only the CNN algorithm has been employed for performing the classification of credit card fraud detection dataset. Additionally to balance the dataset we utilized two balancing techniques namely SMOTE and ADASYN. The figure 9 demonstrates the accuracy of the CNN with and with handling the class imbalance problem.

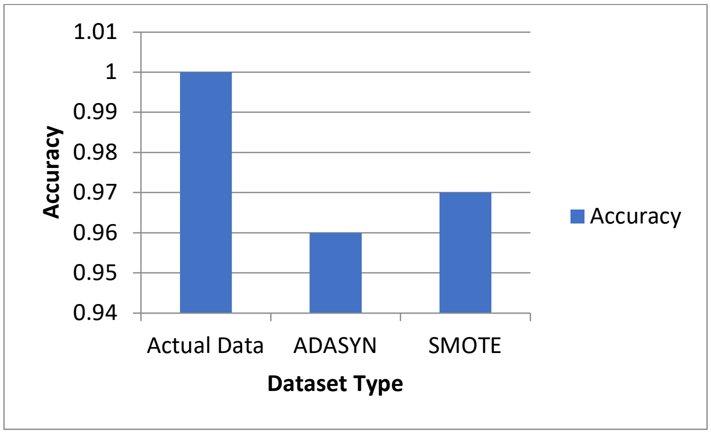

Figure 9: Accuracy of CNN with handling the class imbalance problem

According to the obtained accuracy of the model after correction of the dataset in terms of class imbalance problem, the accuracy of the same classifier is changed. However, the accuracy of the model is reduced but it can be more authentic as compared to the original data classification, because the key issue of the dataset has been handled with the relevant technique. As the final conclusion of the experimental results we can say the improvement in data quality can change the learning performance of classifier. Additionally, the SMOTE is more appropriate algorithm for handling the class imbalance problem as compared to ADASYN algorithm. However, this technique will enhance the learning performance of the classifiers but the amount of data has increased due to this process and can increase the time and memory usage.

3. Experiment after outlier and overlap handling

Next the same dataset is considered and more refinement on dataset has been done.

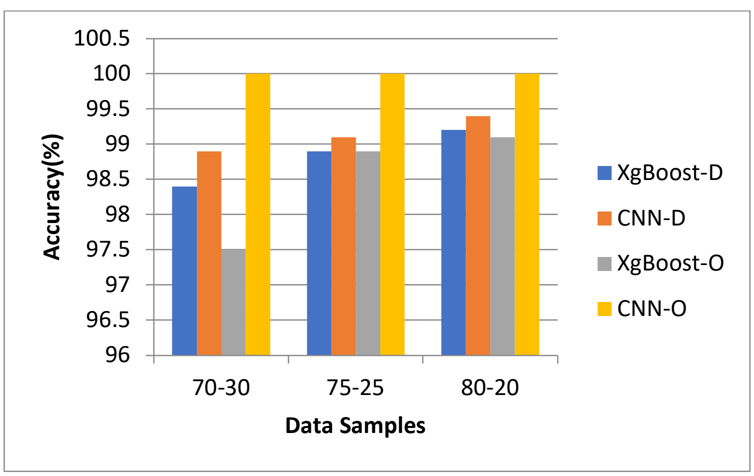

Figure 10: Accuracy after deep clean

The improvement of the data includes:

1. Handling missing values
2. Dimensionality reduction
3. Overlapped attribute handling

4. Outlier detection and removal

Using this deep cleaning technique of credit card fraud detection dataset the size of dataset has dramatically changed. Additionally, a huge difference in classification performance has been observed. The performance here measured and tested on two classifiers and training time. The comparative accuracy of the ML algorithms with original dataset and after deep clean is given using figure 10. According to the obtained results the accuracy of CNN is significantly changed as compared to XgBoost algorithm. Additionally, the accuracy of XgBoost is changing with the use of training and validation ratio. Therefore the deep cleaning and dataset balancing provides similar effect of classifier's learning performance.

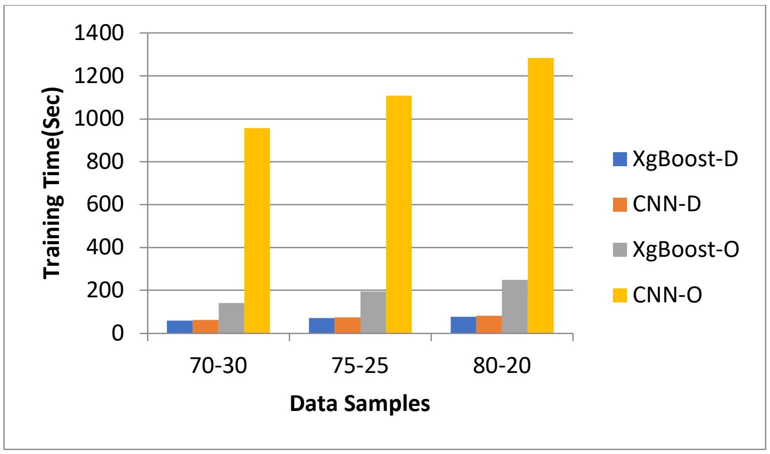

Figure 11: Training time after deep clean

The comparative training time of the classification methods for credit card fraud detection is given in figure 11. According to the performance in terms of training time the XgBoost consumes less amount of time as compared to CNN is both the experimental datasets. However, the processing time of CNN is higher but after deep clean it reduced significantly. In addition, as the training and validation set has been changed the training time of the ML algorithms are being change. The training time is increasing with the size of training samples are increased. However, the training time is higher of CNN but the accuracy is more effective in all the discussed experimental scenarios. Additionally the deep cleaning technique and class imbalance handling will demonstrate the similar effect on data. But the deep clean can reduce the performance of the ML models in terms of time and memory resource use.

VI. CONCLUSIONS

The credit card fraud is a crucial issue for credit card and banking companies. That causes significant losses of money and credibility of banking company. Therefore, it is necessary to keep security intact. However, a number of ML based models are developed recently for handling this issue accurately. But there is a key issue of availability of dataset due to security reasons. In this context, a popular dataset available online has been utilized for experimental study. Initially the original dataset an experiment has been performed and there are seven ML algorithms has been considered namely SVM, CNN, RF, LR, GNB, KNN, and k-means. During the experimental study we have found the K-means, KNN and GNB provide the similar accuracy (98%). Additionally, CNN, SVM, RF and LR provide the higher accurate results

(100%). Additionally it is observed that the dataset is suffering from the class imbalance problem.

Thus, the study is focused on handling the class imbalance problem. In this context, two over sampling techniques are utilized namely ADASYN and SMOTE. Here, we use over sampling technique to avoid the information loss. After over sampling with both the methods we utilize the common classifier CNN for classifying the samples. According to this experimental study we found the SMOTE is more better then the ADASYN technique of over sampling. In addition, we compare the performance with the original dataset classification. During this comparison we have found the accuracy of ML algorithm is reduced but provide the authentic results. Additionally we have found the data also has the higher dimensions and also contains different kinds of noisy contents. Therefore the next study is dedicated to handle noise and dimension of the dataset.

The next work is aimed to handle the noise in terms of Handling missing values, Dimensionality reduction, Overlapped attribute handling and Outlier detection and removal. This mix of data cleaning is named here as the deep cleaning. After deep cleaning we utilized two different classifiers XgBoost and CNN, additionally utilize the original dataset for performing the experiment. The results of the experiment has been recorded in terms of accuracy and training time based on the obtained performance we found the deep cleaning is able to reduce the computational resource consumption in terms of time and memory. In addition, accuracy is remain consistent as the over sampling techniques. therefore deep cleaning method is superior then the over sampling techniques.

REFERENCES

[1] S. Mittal, S. Tyagi, "Chapter 26: Computational Techniques for Real-Time Credit Card Fraud Detection", Handbook of Computer Networks and Cyber Security, Springer Nature Switzerland AG, 2020

[2] G. Sasikala, M. Laavanya, B. Sathyasri, C. Supraja, V. Mahalakshmi, S. S. Sreeja Mole, Jaison Mulerikkal, S. Chidambaranathan, C. Arvind, K. Srihari, and Minilu Dejene, "An Innovative Sensing Machine Learning Technique to Detect Credit Card Frauds in Wireless Communications", Hindawi Wireless Comm. and Mobile Computing, Article ID 2439205, 12 pages, 2022

[3] I. Benchaji, S. Douzi, B. E. Ouahidi, "Credit Card Fraud Detection Model Based on LSTM Recurrent Neural Networks", Journal of Advances in Information Technology Vol. 12, No. 2, May 2021

[4] A. Mujumdar, Dr. Vaidehi V, "Diabetes Prediction using Machine Learning Algorithms", Procedia Computer Science, 165, 292–299, 2019

[5] N. Rtaylia, N. Enneya, "Selection Features and Support Vector Machine for Credit Card Risk Identification", Procedia Manufacturing, 46, 941–948, 2020

[6] P. K. Sadineni, "Detection of Fraudulent Transactions in Credit Card using Machine Learning Algorithms", Proceedings of the Fourth International Conference on I-SMAC, IEEE, 2020

[7] O. Adepoju, J. Wosowei, S. lawte, H. Jaiman, "Comparative Evaluation of Credit Card Fraud Detection Using Machine Learning Techniques", Global Conference for Advancement in Technology, IEEE, 2019

[8] D. Varmedja, M. Karanovic, S. Sladojevic, M. Arsenovic, A. Anderla, "Credit Card Fraud Detection - Machine Learning methods", 18th International Symposium Infoteh-Jahorina, IEEE, 20-22 March 2019

[9] A. M. Rahat, A. Kahir, A. K. M. Masum, "Comparison of Naive Bayes and SVM Algorithm based on Sentiment Analysis Using Review Dataset", 8th International Conference on System Modeling & Advancement in Research Trends, IEEE, 22nd–23rd November, 2019

[10] S. Kiran, J. Guru, R. Kumar, N. Kumar, D. Katariya, M. Sharma, "Credit card fraud detection using Naïve Bayes model based and KNN classifier", International Journal of Advance Research, Ideas and Innovations in Technology, Volume 4, Issue 3, 2018

[11] I. Sadgali, N. Sael, N. Sael, "Fraud detection in credit card transaction using neural networks", CASABLANCA, Morocco, Association for Computing Machinery SCA, Oct 2–4, 2019

[12] T. H. Lin, J. R. Jiang, "Credit Card Fraud Detection with Autoencoder and Probabilistic Random Forest", Mathematics, 9, 2683, 2021.

[13] M. R. Dileep, A. V. Navaneeth, M. Abhishek, "A Novel Approach for Credit Card Fraud Detection using Decision Tree and Random Forest Algorithms", Proceedings of the Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), IEEE, 2021

[14] Y. Chen, R. Zhang, "Research on Credit Card Default Prediction Based on k-Means SMOTE and BP Neural Network", Hindawi Complexity, Article ID 6618841, 13 pages, 2021

[15] P. Shanmugapriya, R. Shupraja, V. Madhumitha, "Credit Card Fraud Detection System Using CNN", International Journal for Research in Applied Science & Engineering Technology, Volume 10 Issue III Mar 2022