Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 165 (2019) 631–641

Procedia Computer Science

www.elsevier.com/locate/procedia

INTERNATIONAL CONFERENCE ON RECENT TRENDS IN ADVANCED COMPUTING 2019, ICRTAC 2019

Credit Card Fraud Detection using Machine Learning Algorithms

Vaishnavi Nath Dornadula[a*], Geetha S[a]

[a]Vellore Institute of Technology, Chennai-600127, India

Abstract

Credit card frauds are easy and friendly targets. E-commerce and many other online sites have increased the online payment modes, increasing the risk for online frauds. Increase in fraud rates, researchers started using different machine learning methods to detect and analyse frauds in online transactions. The main aim of the paper is to design and develop a novel fraud detection method for Streaming Transaction Data, with an objective, to analyse the past transaction details of the customers and extract the behavioural patterns. Where cardholders are clustered into different groups based on their transaction amount. Then using sliding window strategy [1], to aggregate the transaction made by the cardholders from different groups so that the behavioural pattern of the groups can be extracted respectively. Later different classifiers [3],[5],[6],[8] are trained over the groups separately. And then the classifier with better rating score can be chosen to be one of the best methods to predict frauds. Thus, followed by a feedback mechanism to solve the problem of concept drift [1]. In this paper, we worked with European credit card fraud dataset.

Keywords: Card-Not-Present frauds, Card-Present-Frauds, Concept Drift

*Corresponding Author. Tel.: +91 9632474839

Email Address: d.vaishnavinath@gmail.com

10.1016/j.procs.2020.01.057

1. Introduction

Credit card generally refers to a card that is assigned to the customer (cardholder), usually allowing them to purchase goods and services within credit limit or withdraw cash in advance. Credit card provides the cardholder an advantage of the time, i.e., it provides time for their customers to repay later in a prescribed time, by carrying it to the next billing cycle.

Credit card frauds are easy targets. Without any risks, a significant amount can be withdrawn without the owner's knowledge, in a short period. Fraudsters always try to make every fraudulent transaction legitimate, which makes fraud detection very challenging and difficult task to detect.

In 2017, there were 1,579 data breaches and nearly 179 million records among which Credit card frauds were the most common form with 133,015 reports, then employment or tax-related frauds with 82,051 reports, phone frauds with 55,045 reports followed by bank frauds with 50,517 reports from the statics released by FTC [10].

Fig. 1: Taxonomy for Frauds

With different frauds mostly credit card frauds, often in the news for the past few years, frauds are in the top of mind for most the world's population. Credit card dataset is highly imbalanced because there will be more legitimate transaction when compared with a fraudulent one.

As advancement, banks are moving to EMV cards, which are smart cards that store their data on integrated circuits rather than on magnetic stripes, have made some on-card payments safer, but still leaving card-not-present frauds on higher rates.

According to 2017 [10], the US Payments Forum report, criminals have shifted their focus on activities related to CNP transactions as the security of chip cards were increased. Fig 2, shows the number of CNP frauds cases that were registered in respective years.

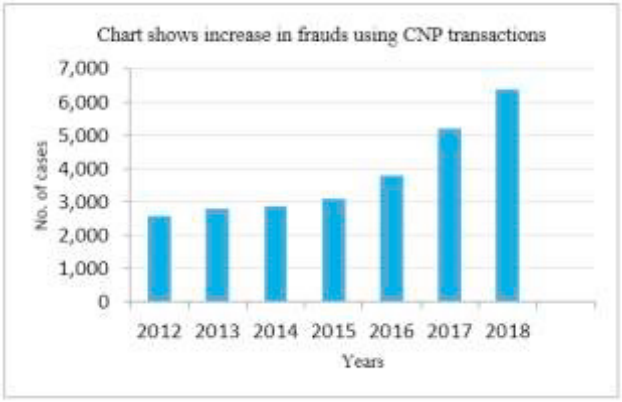

Fig. 2: Frauds Using Card Not Present Transaction

Even then there are chances for thieves to misuse the credit cards. There are many machine learning techniques to overcome this problem.

2. Literature Survey

Multiple Supervised and Semi-Supervised machine learning techniques are used for fraud detection [8], but we aim is to overcome three main challenges with card frauds related dataset i.e., strong class imbalance, the inclusion of labelled and unlabelled samples, and to increase the ability to process a large number of transactions.

Different Supervised machine learning algorithms [3] like Decision Trees, Naive Bayes Classification, Least Squares Regression, Logistic Regression and SVM are used to detect fraudulent transactions in real-time datasets. Two methods under random forests [6] are used to train the behavioural features of normal and abnormal transactions. They are Random-tree-based random forest and CART-based. Even though random forest obtains good results on small set data, there are still some problems in case of imbalanced data. The future work will focus on solving the above-mentioned problem. The algorithm of the random forest itself should be improved.

Performance of Logistic Regression, K-Nearest Neighbour, and Naïve Bayes are analysed on highly skewed credit card fraud data where Research is carried out on examining meta-classifiers and meta-learning approaches in handling highly imbalanced credit card fraud data.

Through supervised learning methods can be used there may fail at certain cases of detecting the fraud cases. A model of deep Auto-encoder and restricted Boltzmann machine (RBM) [2] that can construct normal transactions to find anomalies from normal patterns. Not only that a hybrid method is developed with a combination of Adaboost and Majority Voting methods [4].

3. Proposed Method

Card transactions are always unfamiliar when compared to previous transactions made the customer. This

unfamiliarity is a very difficult problem in real-world when are called concept drift problems [1]. Concept drift can be said as a variable which changes over time and in unforeseen ways. These variables cause a high imbalance in data. The main aim of our research is to overcome the problem of Concept drift to implement on real-world scenario. Table 1, [1] shows basic features that are captured when any transaction is made.

Table 1: Raw features of credit card transactions

Attribute name	Description
Transaction id	Identification number of a transaction
Cardholder id	Unique Identification number given to the cardholder
Amount	Amount transferred or credited in a particular transaction by the customer
Time	Details like time and date, to identify when the transaction was made
Label	To specify whether the transaction is genuine or fraudulent

3.1 Dataset Description

The dataset [11] contains transactions made by a cardholder in a duration in 2 days i.e., two days in the month of September 2013. Where there are total 284,807 transactions among which there are 492 i.e., 0.172% transactions are fraudulent transactions. This dataset is highly unbalanced. Since providing transaction details of a customer is considered to issue related to confidentiality, therefore most of the features in the dataset are transformed using principal component analysis (PCA). V1, V2, V3,..., V28 are PCA applied features and rest i.e., 'time', 'amount' and 'class' are non-PCA applied features, as shown in table 2.

Table 2: Attributes of European dataset

S. No.	Feature	Description
1.	Time	Time in seconds to specify the elapses between the current transaction and first transaction.
2.	Amount	Transaction amount
3.	Class	0 - not fraud 1 – fraud

Fig. 3 shows the correlation matrix of the dataset. This matrix explains that attribute class is independent of both the amount and time of the transaction was made. It is even clear from the matrix, the class of the transaction is depending on PCA applied attributes.

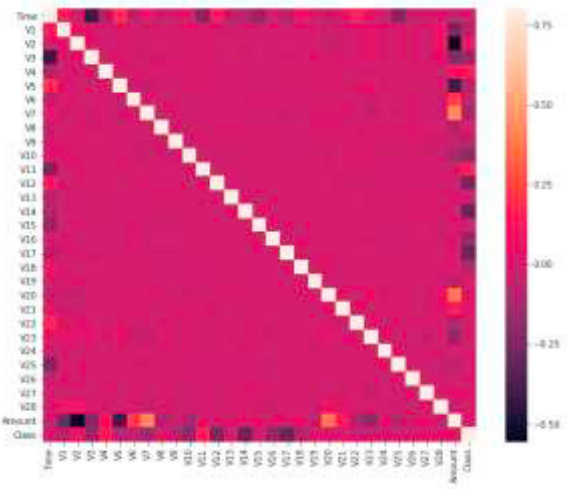

Fig. 3: Correlation Matrix for Attributes (both the X and Y axis show different attributes present in dataset)

4. Methodology

- Firstly, we use clustering method to divide the cardholders into different clusters/groups based on their transaction amount, i.e., high, medium and low using range partitioning.
- Using Sliding-Window method, we aggregate the transactions into respective groups, i.e., extract some features from window to find cardholder's behavioural patterns. Features like maximum amount, minimum amount of transaction, followed by the average amount in the window and even the time elapsed.

Algorithm 1: Algorithm to derive aggregated transaction details and to extract card holder features using sliding window technique.

Input: id of the customer holding a card, a sequence of transactions t and window size w;

Output: Aggregated transactions details and features of cardholder genuine or fraud;

```
l: length of T
Genuine= [];
Fraud= [];
For i in range 0 to l-w+1:
        T: [];
        /* sliding window features*/
        For j in range i+w-1:
```

*/*Add the transaction to window */*

$T=T+t_j^{id}$;

End

/ features extraction related to amount */*

$a_{i1}=MAX\_AMT(T_i)$;

$a_{i2}=MIN\_AMT(T_i)$;

$a_{i3}=AVG\_AMT(T_i)$;

$a_{i4}=AMT(T_i)$;

For j in range i+w-1:

/ Time elapse */*

$x_{i=}\ Time(t_j)-Time(t_{j-1})$

End

$X_i= (a_{i1}, a_{i2,}a_{i3,}a_{i4,}a_{i5,})$;

$Y= LABEL(T_i)$;

/ classifying a transaction into fraud or not */*

if $Y_i=0$ *then*

$Genuine =Genuine\ U\ X_i$;

Else

$Fraud =Fraud\ U\ X_i$;

End

- Every time a new transaction is fed to the window the old once are removed and step-2 is processed for each group of transactions. (Algorithm for Sliding-Window based method to aggregate are referred from [1]).
- After pre-processing, we train different classifiers on each group using the cardholders behavioural patterns in that group and extract fraud features. Even when we apply classifiers on the dataset, due to imbalance (shown in fig 4) in the dataset, the classifiers do not work well on the dataset.

Fig. 4: Transaction Class Distribution in Dataset

- Thus, we perform SMOTE (Synthetic Minority Over-Sampling Technique) operation on the dataset.
- Oversampling does not provide any good results.
- Thus, there are two different ways of dealing with imbalance dataset i.e., consider Matthew Coefficient Correlation of the classifier on the original dataset or we make use of one-class classifiers.
- Finally, the classifier that is used for training the group is applied to each cardholder in that group. The classifier with highest rating score is considered as cardholder's recent behavioural pattern.
- Once the rating score [1] is obtained, now we append a feedback system, wherein the current transaction and updated rating score are given back to the system (for further comparison) to solve the problem of concept drift.

Algorithm 2: Algorithm to update the rating score of the classifier to find the accurate the model is.

Input: id of the cardholder and a pervious and a current transaction.
Output: Rating score of the model after every transaction.

T: current transaction with w-1 transaction from window.
C: represents the classifier
Label: true value of the incoming/current transaction.
K: total of transactions processed by model.
If the predicted value ≠ label and label==0 then,
    For i in range (0, K):
        If the predicted value ≠ label then,
        $rs_i = rs_i - 1;$
    Else
        $rs_i = rs_i + 1;$
End

4.1 Formula

In our proposed system we use the following formulae to evaluate, accuracy and precision are never good parameters for evaluating a model. But accuracy and precision are always considered as the base parameter to evaluate any model.

The Matthews Correlation Coefficient (MCC) is a machine learning measure which is used to check the balance of the binary (two-class) classifiers. It takes into account all the true and false values that is why it is generally regarded as a balanced measure which can be used even if there are different classes,

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{3}$$

TP- True Positive
TN- True Negative
FP- False Positive
FN- False Negative

5. Experimental Results

We have experimented few models on original as well as SMOTE dataset. The results are tabulated, which shows great differences in accuracy, precision and MCC as well. We even used one-class SVM which can be best used for binary class datasets. Since we have 2 classes in our dataset we can use one-class SVM as well.

Table 3, shows the results on the dataset before applying SMOTE and fig 5, shows the same results graphically.

Table 3: Accuracy, Precision and MCC values before applying SMOTE,

Methods	Accuracy	Precision	MCC
Local Outlier factor	0.8990	0.0038	0.0172
Isolation forest	0.9011	0.0147	0.1047
Support vector machine	0.9987	0.7681	0.5257
Logistic regression	0.9990	0.875	0.6766
Decision tree	0.9994	0.8854	0.8356
Random forest	0.9994	0.9310	0.8268

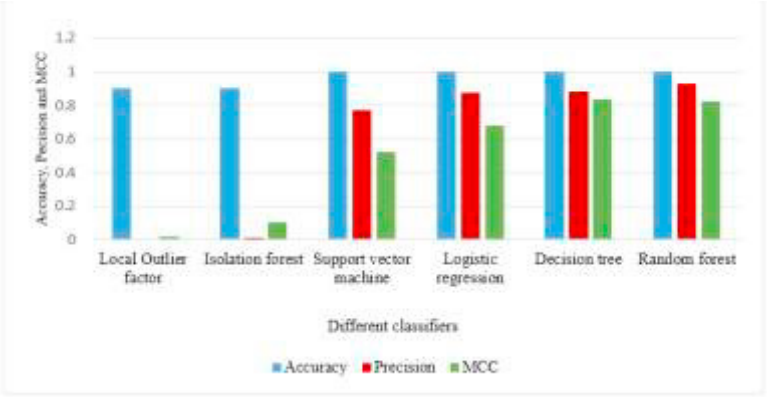

Fig 5: chart showing results on original dataset

One-Class SVM

Accuracy: 0.7009

Precision: 0.7015

Table 4, shows the results on the dataset after applying SMOTE and fig 6, shows the same results graphically.

Table 4: Accuracy, Precision and MCC values after applying SMOTE,

Methods	Accuracy	Precision	MCC
Local Outlier factor	0.4582	0.2941	0.1376
Isolation forest	0.5883	0.9447	0.2961
Logistic regression	0.9718	0.9831	0.9438
Decision tree	0.9708	0.9814	0.9420
Random forest	0.9998	0.9996	0.9996

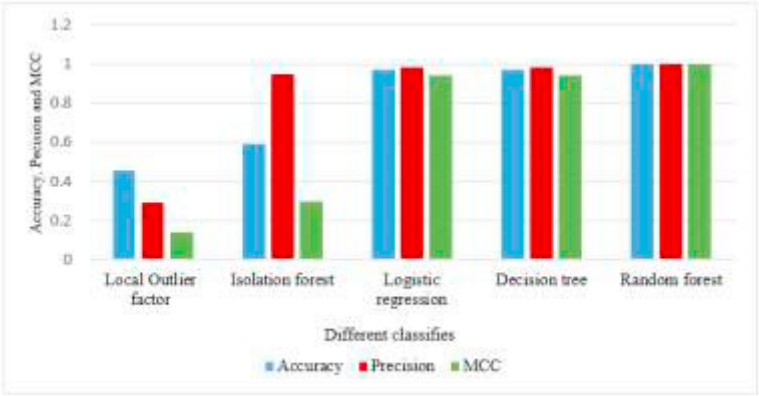

Fig 6: chart showing results on updated dataset

Fig 7, shows the comparison between the values of MCC on dataset before and after applying SMOTE.

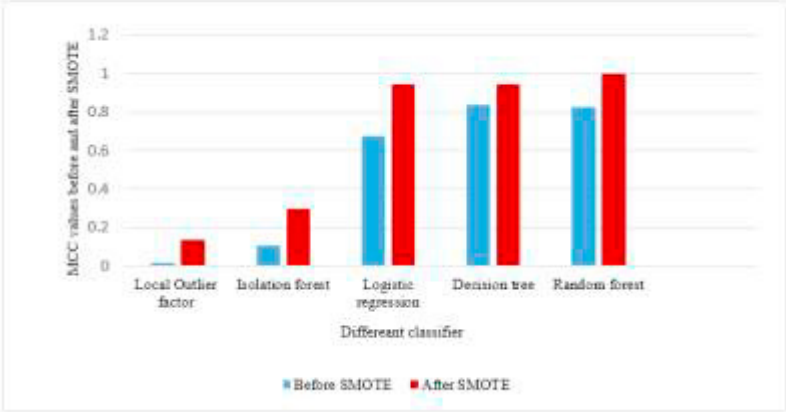

Fig 7: MCC parameter comparison between original and updated dataset

6. Conclusion

In this paper we developed a novel method for fraud detection, where customers are grouped based on their transactions and extract behavioural patterns to develop a profile for every cardholder. Then different classifiers are applied on three different groups later rating scores are generated for every type of classifier. This dynamic changes in parameters lead the system to adapt to new cardholder's transaction behaviours timely. Followed by a feedback mechanism to solve the problem of concept drift. We observed that the Matthews Correlation Coefficient was the better parameter to deal with imbalance dataset. MCC was not the only solution. By applying the SMOTE, we tried balancing the dataset, where we found that the classifiers were performing better than before. The other way of handling imbalance dataset is to use one-class classifiers like one-class SVM. We finally observed that Logistic regression, decision tree and random forest are the algorithms that gave better results.

REFERENCES

[1] Jiang, Changjun et al. "Credit Card Fraud Detection: A Novel Approach Using Aggregation Strategy and Feedback Mechanism." *IEEE Internet of Things Journal* 5 (2018): 3637-3647.

[2]Pumsirirat, A. and Yan, L. (2018). Credit Card Fraud Detection using Deep Learning based on Auto-Encoder and Restricted Boltzmann Machine. International Journal of Advanced Computer Science and Applications, 9(1).

[3]Mohammed, Emad, and Behrouz Far. "Supervised Machine Learning Algorithms for Credit Card Fraudulent Transaction Detection: A Comparative Study." IEEE Annals of the History of Computing, IEEE, 1 July 2018, doi.ieeecomputersociety.org/10.1109/IRI.2018.00025.

[4]Randhawa, Kuldeep, et al. "Credit Card Fraud Detection Using AdaBoost and Majority Voting." IEEE Access, vol. 6, 2018, pp. 14277–14284., doi:10.1109/access.2018.2806420.

[5]Roy, Abhimanyu, et al. "Deep Learning Detecting Fraud in Credit Card Transactions." 2018 Systems and Information Engineering Design Symposium (SIEDS), 2018, doi:10.1109/sieds.2018.8374722.

[6]Xuan, Shiyang, et al. "Random Forest for Credit Card Fraud Detection." 2018 IEEE 15th International Conference on Networking, Sensing and Control (ICNSC), 2018, doi:10.1109/icnsc.2018.8361343.

[7]Awoyemi, John O., et al. "Credit Card Fraud Detection Using Machine Learning Techniques: A Comparative Analysis." 2017 International Conference on Computing Networking and Informatics (ICCNI), 2017, doi:10.1109/iccni.2017.8123782.

[8]Melo-Acosta, German E., et al. "Fraud Detection in Big Data Using Supervised and Semi-Supervised Learning Techniques." 2017 IEEE Colombian Conference on Communications and Computing (COLCOM), 2017, doi:10.1109/colcomcon.2017.8088206.

[9] http://www.rbi.org.in/Circular/CreditCard

[10] https://www.ftc.gov/news-events/press-releases/2019/02/imposter-scams-top-complaints-made-ftc-2018

[11] https://www.kaggle.com/mlg-ulb/creditcardfraud

[12] https://www.kaggle.com/uciml/default-of-credit-card-clients-dataset

[13] https://www.kaggle.com/ntnu-testimon/paysim1/home