IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Revue d'Intelligence Artificielle
Vol. 36, No. 4, August, 2022, pp. 621-628
Journal homepage: http://iieta.org/journals/ria

Online Transaction Fraud Detection Using Efficient Dimensionality Reduction and Machine Learning Techniques

Rathan Kumar Chenoori[1*], Radhika Kavuri[2]

[1] Department of Computer Science and Engineering, Osmania University, Telangana 500007, India
[2] Department of Information Technology, CBIT, Telangana 500075, India

Corresponding Author Email: rathanoucse@gmail.com

https://doi.org/10.18280/ria.360415

Received: 28 June 2022
Accepted: 19 August 2022

Keywords:
machine learning, XGBoost, fraud detection, PCA, EDA

ABSTRACT

In recent years, there has been a rapid increase in the number of online transactions. Substantial growth has been reported in e-commerce and e-governance in the past few years. Due to this the number of people using online payment methods has also increased. This has led to an exponential rise in the number of transactions that happen every day. This increase in online transactions has further led to an increase in the number of frauds in the transactions. There is an ever-growing need to detect these fraudulent transactions as early as possible so that appropriate actions could be taken and losses due to these frauds could be minimized. This work proposes machine learning models which could use the previously known data and try to predict frauds based on information learned through the old data. We propose a statistical based dimensionality reduction technique and various machine learning models were tried for classification purpose. We experimented our proposed method on IEEE-CIS Fraud Detection dataset and the best results were obtained on the XGBoost model which is demonstrated in this paper.

1. INTRODUCTION

In the last two decades [1-3] the digital economy has grown very fast. People who were earlier totally dependent on cash transactions have also started using online payments. The ease of use, hassle-free and smooth user experience and time-efficient use of e-commerce [4], online banking, e-governance [5, 6] etc. have facilitated the shift from cash-based transactions to online transactions. All this is aided by the advancements in technology that have given every household, whether urban or rural, access to the internet.

As everything is moving online, the number of online and cashless transactions is increasing. This shift was further aided by the pandemic. This increase along with great benefits poses some challenges as well. One of the major challenges is the number of frauds that happen in online transactions. As the quantity of online transactions grows, so does the number of online frauds. This calls for methods that could detect fraudulent transactions as soon as possible. This will allow the concerned authorities to take necessary measures to minimize the loss to public and private entities as well as people at large.

Machine learning algorithms [7] try to predict the outcome or provide us with the details about a data sample based on previous inferences. Based on the data available from prior transactions, the same can be used to forecast if a transaction is fraudulent or not. It is a supervised classification task [8, 9], wherein the aim is to detect and flag fraudulent transactions. As the system should work in a real environment and so would require fast responses ML techniques would be ideal as compared to deep learning techniques. This is so because the deep learning algorithm tends to take time while training as well as while giving inferences.

1.1 Contribution

Due to a large number of features and attributes, exploratory data analysis and feature selection were carried out to reduce the load and also gain an understanding of which features are more important.

Contribution of the paper is as follows:

i. In this, we proposed a dimensionality reduction technique before classification. Exploratory data analysis allowed converting the dataset into a usable format for the model. Detected dataset was used for training and testing the model.

ii. We adopted XGBoost [10] which is known to perform well in classification tasks. This work tries to explore the use of XGBoost for online fraud detection on IEEE-CIS Fraud dataset.

iii. Finally, the goal of this work is to create a model which is efficient enough and provides accurate results. This would allow the model to be used in real-world applications.

2. LITERATURE SURVEY AND METHODOLOGIES

In this study, the main aim is to detect fraudulent transactions using credit cards with the help of ML algorithms and deep learning algorithms. Rahul etal. [7] used data mining techniques to investigate various counterfeit transaction methods used in credit card frauds and try to identify them. It looks into the numerous approaches that can be used to identify credit card fraud before pointing out the flaws in present systems. This study was done from the data provided by a large Brazilian credit card issuer.

Aditya Oza [11] presented "Fraud Detection using Machine Learning" that applies three different Machine Learning techniques with a focus on the problem of online payment fraud detection. The dataset (Paysim) used here was obtained from Kaggle and it is a collection of simulated mobile-based payment transactions. They performed Principal Component Analysis (PCA) to project the variableness of data in the 2-dimensional space. Finally, the effectiveness in detecting fraudulent transactions of both the models used in this paper was compared.

In 2019, Saputra et al. [12] published a research paper titled "Fraud Detection using Machine Learning in e-Commerce" tries to analyze the best machine learning algorithm which would be suitable for fraud detection in online transactions. It makes use of Random Forest, Naive Bayes, Decision Tree and Neural Networks in this study. The dataset used was unbalanced. So, the authors made use of the Synthetic Minority Over-sampling Technique (SMOTE) process to generate a balanced dataset for the study. It was found out that the Neural Network performed the best with 96% accuracy.

In 2019, Venu et al. [13] shared their work "Analysis of Credit Card Fraud Data using PCA" which used PCA and K-Means clustering algorithm to detect credit card frauds done in the years 2001 to 2016. The graph depicts the overall number of fraud transactions carried out by males, females, and others in an Indian state. Fraudulent transactions carried out by men are depicted in green, fraudulent transactions by women are depicted in red, and fraudulent transactions by others are depicted in blue.

According to Maniraj et al. [14] "Credit Card Fraud Detection using Machine Learning and Data Science", is firstly passed the dataset through a local outlier factor and then through an isolation forest algorithm [15-17]. Then the authors applied the technique on the dataset obtained from a German bank in 2006. To maintain the anonymity of the data points of the dataset, the German bank provided only a summary of the transactions. The statistical results obtained after applying the Local Outlier Factor and the Isolation.

Pumsirirat et al [18] and Yang et al. [19] proposed a no-super-parametric improvement to AdaBoost [20] and which is used for fraud detection. Although the performance of this AdaBoost without super-parameters is a bit lesser than existing AdaBoost, it still outperforms others including the original AdaBoost and other existing improvements of AdaBoost.

3. PROPOSED METHOD FOR IEEE-CIS FRAUD DETECTION DATASET

The researchers from the IEEE Computational Intelligence Society (IEEE-CIS) [21] wanted to prevent cases of online transaction frauds. This would save millions of dollars each year to the users who make online transactions. In an attempt to improve online transaction fraud detection, they partnered with Vesta Corporation, which is one of the world's leading payment service companies. Together they provided a public dataset that comprises authentic transactions as well as fraud transactions. This dataset can be used to determine whether or not an online transaction is fraudulent, and this can be verified by making use of a binary target variable "isFraud". Details of dataset are provided in Table 1 and Table 2.

Table 1. Identity dataset features

Feature	Description
TransactionID	ID of transaction
DeviceType	device type entered.
DeviceInfo	device information.
id_1 - id_38	masked features corresponding to the detail of the open session and user logging.

Table 2. Transaction dataset features

Feature	Description
TransactionID	ID of transaction
ifFraud	details about prelabelled transanction
TransactionDT	timedelta from a given reference datetime
TransactionAMT	transaction payment amount in USD.
ProductCD	product code, the product for each transaction.
card1 - card6	6payment card information, such as card type, card category, issue bank, country
addr	billing country (addr2) and billing region (addr1).
dist	distance
P_ and R_ email domain	purchaser and recipient email domain
C1-C14	counting, such as how many addresses are found to be associated with the payment card, etc.
D1-D15	timedelta, such as days between previous transaction, etc.
M1-M9	match, such as names on card, address
Vxxx	Vesta engineered rich features, including ranking, counting, and other entity relations.

3.1 Proposed framework

In order to identity the fraud transactions in this work we initially perform the EDA techniques to find the relationship among the features avalible in the dataset represented in Table 1, Table 2, then the new Dimensionality reduction method identifies and remove unnecessary features from dataset and finally trained a model with train dataset and evaluated the performance of model using test dataset. The framework of our model is presented in Figure 1.

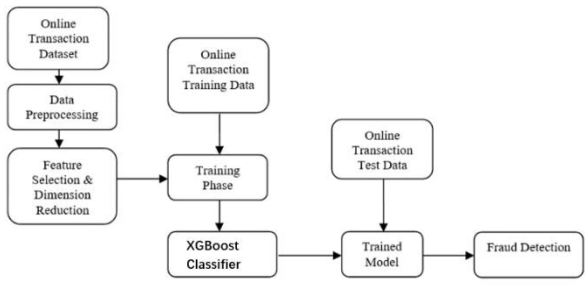

Figure 1. Framework model

3.2 Exploratory data analysis (EDA) and feature selection

We would initially conduct an Exploratory Data Analysis (EDA) [22, 23] on the "identity" and "transaction" datasets, to get a general understanding of the entire dataset and the relationship between them. Exploratory Data Analysis (EDA) can be performed using a variety of methods, such as

histograms, heatmaps, correlation matrices, box plots, bar charts (grouped and ungrouped).

This allows us to reach the process of feature selection, in which finally we use only those variables that seem relevant to getting a prediction value. The inclusion of unnecessary and irrelevant variables can affect the performance of the machine learning model. It helps us to reduce the dimensionality of the final dataset to be used, which in turn reduces the computational cost and complexity of the machine learning model. It also prevents the overfitting of the model.

In the "identity" dataset, the variables "id_01" - "id_11" are continuous variables, while all the remaining variables are categorical. The data present inside this subset of the dataset represents information about the identity of the user and are related to the transactions. Now, coming over to the "transaction" dataset, there are about 20 categorical variables present in this subset of the dataset. Like the "identity" subset, critical information about the transactions is also masked to conceal their true meaning and usage.

Then for the categorical variables with low cardinality features are label encoded, while the higher cardinality features are encoded by making use of target encoding using the One Hot Encoding method.

3.3 Dimension reduction phase

In this subsection, we propose an efficient dimensionality reduction of features with the references [24, 25]. We have a corpus of d training documents, each of which is described by n features. Our main goal is to choose a small number l of features, where $l \leq n$, such that, by using only those l features, we can obtain good classification quality, both in theory and in practice, when compared to using the full set of n features. The algorithmic steps are as follows:

Procedure (X, Y) –

1. Consider, X: Be set of conditional variables; Y: Be the decision variable
2. def Heuristic as, Y = [y_1] andX = [x_1, $x_2 \cdots x_m$]
$$H(X \mid Y) = -\mathrm{E}[\log(\, p(x \mid y))]$$
3. $L \leftarrow \phi$
4. do
5. $M \leftarrow L$
6. For all $x \in (X - L)$
7. if $H(L \cup \{x\} \mid Y) < H(M \mid Y)$
8. $M \leftarrow (L \cup \{x\})$
9. $L \leftarrow M$
10. Run up to $H(L \mid Y) < H(X \mid Y)$
11. Return L

The above dimensionality reduction phase works as, it initially takes' m' number of X variables and one Y decision variable. Then Line 2 is used to calculates conditional entropy of X given Y. The Line 3 represents empty list which is used to store reduced features. Next Line 4 to Line 10 are repeated by calculating conditional entory value on each feature of X to identified feature to List L and finally it ends while the entopy of identified features L value on Y exceeds the entropy value of X on given Y.Then it finally returns L, which is the list of features identified by our proposed dimentionality reduction method.

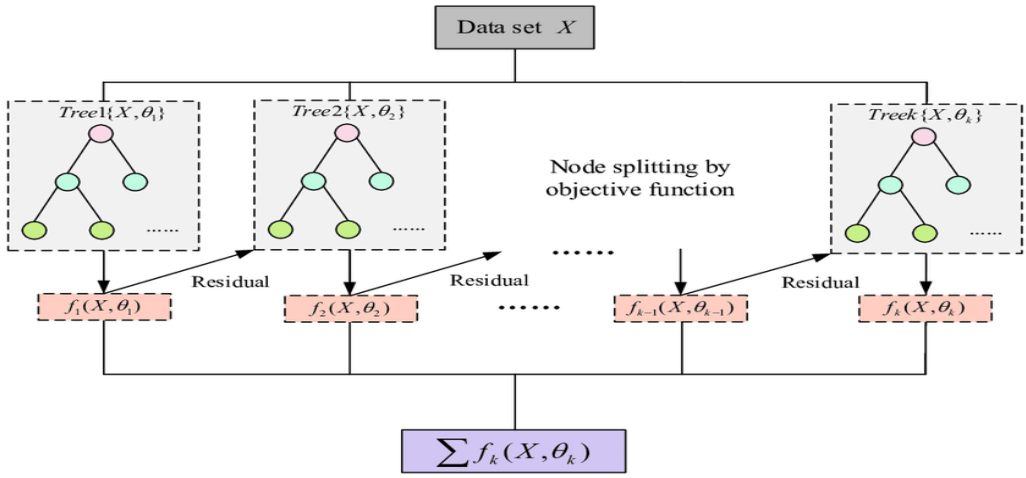

Figure 2. Framework of XGBoost model

3.4 XGBoost model

XGBoost (Extreme Gradient Boosting) is a machine learning algorithm that optimizes gradient descent boosting, by using gradient boosted decision trees. Figure 2 represents the evaluation of XGBoost model. It is an advanced machine learning algorithm that is capable of dealing with data that has higher degree of irregularity in it. XGBoost is an implementation of Gradient Boosted decision trees.

Eq. (1) represents the prediction scores of each individual decision tree mathematically as,

$$\hat{y} = \sum_{i=1}^{k} f_k(x_i), f_k \in F \tag{1}$$

where, k is the number of trees.

For the above step, Eq. (2) represents the objective function at n iterations as:

$$obj(\theta) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{n} \Omega(f_k) \tag{2}$$

where, first term l is the training loss function such as square loss or logistic loss computed with values of $(y_i, \hat{y}_i)$, and the second is the regularization parameter.

Eq. (3) represents the additive strategy applying process for minimization as,

$$\hat{y}_i^{(t)} = \sum_{k=1}^{t} f_k(x_i) = \hat{y}_i^{(t-1)} + f_t(x_i) \tag{3}$$

where, $\hat{y}_i^{(0)} = 0$.

For the above step, Eq. (4) termed the objective function as

$$obj^t = \sum_{i=1}^{n} [2(\hat{y}_i^{(t-1)} - y_i) f_t(x_i) + f_t(x_i)^2] + \Omega(f_k) + C \tag{4}$$

where, C is constant.

Now, Eq. (5) termed second order expansion after applying taylor series:

$$obj^t = \sum_{i=1}^{n} [l(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2} h_i {f_t}^2(x_i)] + \Omega(f_k) + C \tag{5}$$

where, g_i and h_i are defined in Eq. (6) and Eq. (7) as:

$$g_i = \partial_{\hat{y}_i^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)}) \tag{6}$$

$$h_i = \partial^2_{\hat{y}_i^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)}) \tag{7}$$

Eq. (8) termed simplifying form after removing the constant:

$$\sum_{i=1}^{n} [g_i f_t(x_i) + \frac{1}{2} h_i {f_t}^2(x_i)] + \Omega(f_k) \tag{8}$$

Eq. (9) termed the definition of the model:

$$f\ (x_i) = w_{q(x)}, w \in R^T, R^d \to \{1,2,..,T\} \tag{9}$$

Here, $w$is the vector of scores.

The regularization term is then defined in Eq. (10) as:

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} {w_j}^2 \tag{10}$$

Now, our objective function showed in Eq. (11) and Eq. (12) as:

$$obj^t = \sum_{j=1}^{T} \left[(\sum_{i \in I_J} g_i) w_j + \frac{1}{2} (\sum_{i \in I_J} h_i + \lambda) w_j^2 \right] + \gamma T \tag{11}$$

Now, we simplify the above expression:

$$obj^t = \sum_{j=1}^{T} \left[G_j w_j + \frac{1}{2} (H_j + \lambda) w_j^2 \right] + \gamma T \tag{12}$$

where, $G_j = \sum_{i \in I_J} g_i$, $H_j = \sum_{i \in I_J} h_i$.

Finally from below Eqns.(13)-(15), we get best objective reduction value from the best $w_j$for a given structure q(x), where W_j 's are independent of each other,

$$w_j^* = -\frac{G_j}{H_j + \lambda} \tag{13}$$

$$obj* = -\frac{1}{2} \sum_{j=1}^{T} \frac{{G_j}^2}{H_j + \lambda} + \gamma T \tag{14}$$

$$Gain* = \frac{1}{2} \left[\frac{{G_L}^2}{H_L + \lambda} + \frac{{G_R}^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} \right] - \gamma \tag{15}$$

XGBoost is a classifier with a large number of hyper-parameters. To build a model with these parameter values are so crucial. Therefore, it is necessary to carefully select the parameter values. However, there is no theoretical method to guide the choice of parameters. So with the experience we selected some parameters as a crucial parameters with the following values to build our proposed model. Theyare:

a) subsample = 0.8

The default subsample value is 1. It is used to indicate the part of observations to be randomly selected samples of the decision tree model. Keeping a lower value of "subsample" prevents the model from overfitting, but making these values too small would lead to the underfitting of the model.

b) learning rate (learning_rate) = 0.02

Learning rate is the rate at which the gradient descent algorithm moves towards the minima. We generally want to keep the learning rate high enough that we can reach the minima at a reasonable passage of time, but at the same time we also want to keep it low enough that it doesn't oscillate around the minima. It controls the speed at which the weights and biases of each network are updated during the training process.

c) maximum depth of decision tree (max_depth) = 12

The default maximum depth value is kept as 6. It is generally used to tackle the problem of over-fitting of the machine learning model. And it is not good as depth increases the decision trees often imply that the model has started to overfit on the dataset and it acts as a hard stop on the tree build process.

d) evaluation metric (eval_metric) = auc

It is the evaluation metric that the user wants to be used by the XGBoost model for the validation dataset. Some of the available values are root mean square error (rmse), negative log-likelihood (logloss), multiclass logloss

(mlogloss), area under the curve (auc), mean absolute error (mae) and multiclass classification error rate (merror).

4. EXPERIMENT RESULTS

4.1 Setup and simulation environment

We performed experiment on Linux OS, i7 processor with 3.4 Ghz computing facility.

*Dataset:*The "identification" and "transaction" files in the IEEE-CIS Fraud Detection dataset [26, 27] are linked by using the "TransactionID" variable, which is included in both files. The “identity” file stores information about every unique customer and the “transaction” file stores the information about the transaction, which may be authentic or fraudulent. The memory size of the entire dataset is 1.35 GB. Each of these files is further divided into 2 files each (training and testing set).

4.2 Exploratory data analysis

The dataset was divided into a training set and testing set and the samples in both the sets were 590,540 and 506,691 respectively.

Then we find out which columns in the dataset contain some missing values so that appropriate actions like removing rows containing missing values could be taken. As the data is gathered from earlier transactions it is bound to have certain fields in which some of the data is missing. It is due to the fact that in real-time systems there are instances where all the data is not available every time. The same is true with the “transaction” data. We found out that 195 columns had missing data in some rows.

After looking at various features from the identity subset, the next job is to analyze the transaction dataset. First, we explore the transaction amount attribute and the mean of transaction amount when the transaction is fraudulent and not separately. To view the distribution properly and to eliminate skewness due to large transaction amounts, we apply the log transformation and then see the distribution of the transaction amount attribute, which is present in the dataset as “TransactionAmt”. Then we find out the mean of transaction amount for both classes separately. It is shown in the Figure 3 below that the mean for fraudulent transactions was slightly higher at 149.24 as compared to genuine transactions which are 134.37.

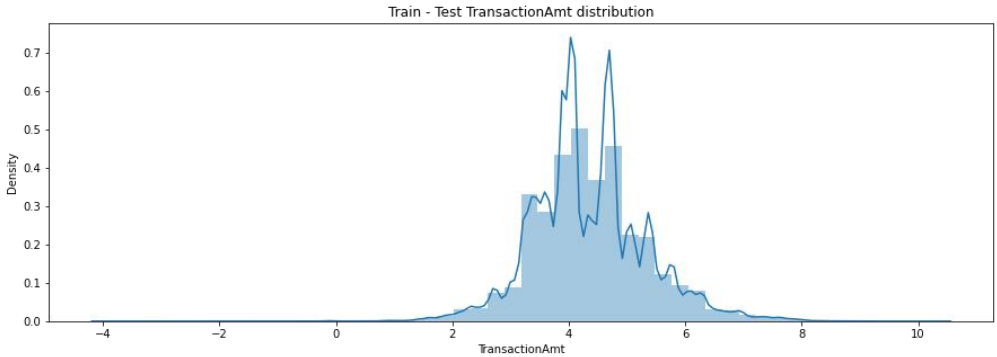

Figure 3. Distribution of transaction amount

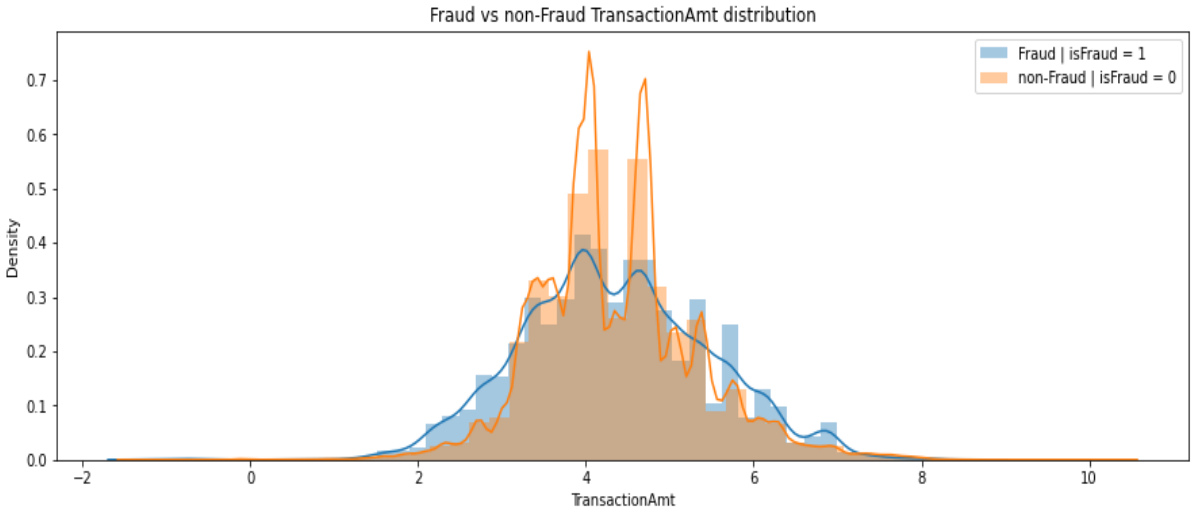

Figure 4. Distribution of transaction amount for both the classes

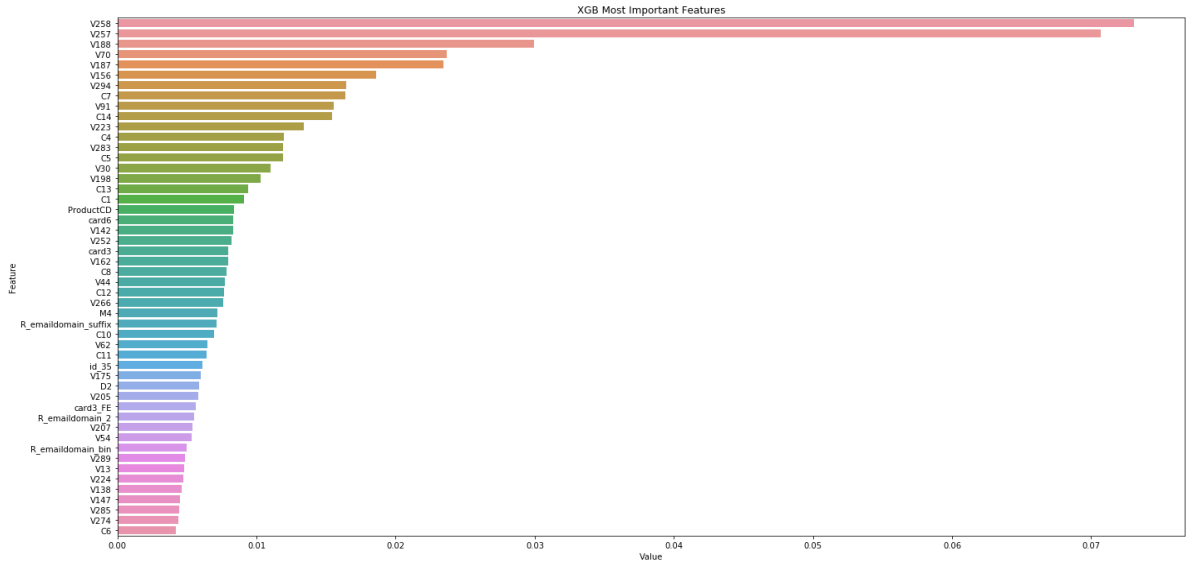

Figure 5. Most important features for the XGBoost model

The distribution of transaction amount for both data from the fraudulent and non-fraudulent transaction classes was then examined and the results are shown in Figure 4.

4.2.1 Feature engineering and feature selection

Feature engineering [28], is the process of using domain knowledge gained from Exploratory Data Analysis (EDA) or prior knowledge to extract certain features from the already existing features which could make the models better. In this work, we use feature engineering to convert e-mail address domains to some well-defined names and then encode them so that they can be used as categorical features in the model.

There was a total of 47 columns that were having NULL values. The columns which had many NULL values do not add much value towards training the models. Hence those columns were dropped from the dataset. After completing all the feature engineering and feature selection we are left with 195 top features that we used to train our XGBoost model with. The following subsection shows the results obtained through the XGBoost model.

```
XGBoost version: 0.90
[0]    validation_0-auc:0.80612
Will train until validation_0-auc hasn't improved in 100 rounds.
[50]   validation_0-auc:0.88063
[100]  validation_0-auc:0.895547
[150]  validation_0-auc:0.908789
[200]  validation_0-auc:0.919616
[250]  validation_0-auc:0.926031
[300]  validation_0-auc:0.9307
[350]  validation_0-auc:0.933865
[400]  validation_0-auc:0.935439
[450]  validation_0-auc:0.936387
[500]  validation_0-auc:0.936519
[550]  validation_0-auc:0.936631
Stopping. Best iteration:
[472]  validation_0-auc:0.936791
```

Figure 6. AUC values during model training

4.3 Results obtained on XGBoost model

We made use of the IEEE-CIS Fraud Detection dataset to train and test the model. We performed Exploratory Data Analysis (EDA) and Feature Selection to exclude irrelevant data variables from the dataset. After running the XGBoost model with the AUC (Area Under Curve) evaluation metric, we obtained the best result as 0.936791, Accuracy of 89% was obtained with good values of precision and F1-score for both classes. Also, among the various features of the dataset, the variable Figure 5 represents "v258" was found to be the most important feature for the XGBoost model. It was found that the XGBoost model also performed pretty well on the given dataset.

Table 3 represents the performance metrics of our experiment compared with other models.

Table 3. Performance metrics for the model

Model	AUC
AggRF+FB	0.84
SMOTE + ML	0.86
ML + AdaBoost	0.88
LightGBM	0.789
XGBoost	0.9367
AggRF+FB	0.84

Figure 6 shows the values of AUC improving through the training process. The best value obtained was 0.936 and the model at this iteration was used for further analysis. From Figure 7 it is observed that XGBoost performs well on AUC and XGBoost is better than previous models like AggRF+FB [1], SMOTE+ML [12], AdaBost [14], LightGBM [21].

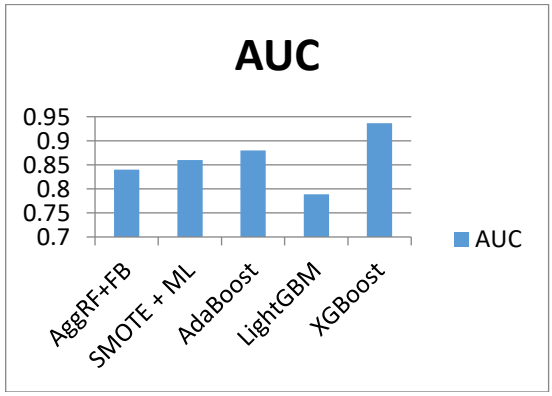

Figure 7. AUC comparison with other models

We tested with Logistic Regression (LR), Support Vector Machine (SVM), Random Forest (RF) [27] and our proposed XGBoost. We obtained results for Logistic regression: 0.84018 (Train AUC), 0.84245 (Test AUC), SVM: 0.88018 (Train AUC), 0.88245 (Test AUC), Random Forest: 0.9030 (Train AUC), 0.8600 (Test AUC), Xgboost: 0.994 (Train AUC), 0.9367 (Test AUC).

Table 4. XGBoost on compared with other methods

Methods / Metrics	LR	SVM	RF	Ada Boost	XGBoost
F1	0.81	0.88	0.78	0.90	**0.92**
Recall	0.78	0.81	0.84	0.87	**0.88**
Precision	0.86	0.91	0.92	0.92	**0.97**
AUC	0.84	0.88	0.86	0.81	**0.9367**

The performance of XGBoost on compared with other methods presented in Table 4 and Figure 8. When using the XGBoost approach, which has the best performance, the best results are attained.

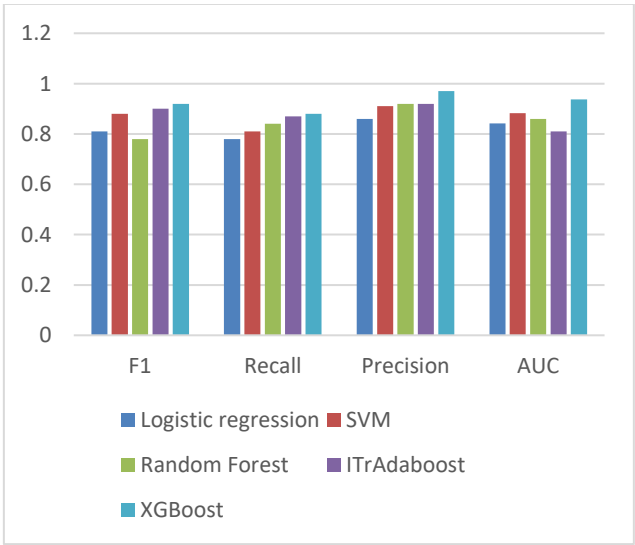

Figure 8. Comparing with other methods

5. CONCLUSION

With the rapid adoption of online payment methods, there has been a substantial increase in the number of online transactions, which in turn has also resulted in an increase in the number of online transaction frauds. This problem has led to the search for a solution that manages to detect such fraudulent transactions within a short span of time. A shorter response time would minimize the cases of fraudulent transactions and would in turn save a lot of people from incurring heavy monetary losses. A variety of research work has been carried out on this subject using different machine learning algorithms. We proposed an efficient dimensionality reduction. We wanted to test the effectiveness of XGBoost, a machine learning technique that has not previously been employed by researchers, on this topic. We made use of the IEEE-CIS Fraud Detection dataset to train and test the model. It was found that the XGBoost model also performed pretty well on the given dataset, and can be adapted for use at a larger scale.

REFERENCES

[1] Jiang, C., Song,J., Liu, G., Zheng, L., Luan, W. (2018). Credit card fraud detection: A novel approach using aggregation strategy and feedback mechanism. IEEE Internet of Things Journal, 5(5): 3637-3647.http://dx.doi.org/10.1109/JIOT.2018.2816007

[2] Srivastava, A., Kundu, A., Sural, S.,Majumdar, A. (2008) Credit card fraud detection using Hidden Markov Model. IEEE Transactions on Dependable and Secure Computing, 5(1): 37-48.http://dx.doi.org/10.1109/TDSC.2007.70228

[3] Randhawa, K., Loo, C.K., Seera, M., Lim, C.P., Nandi, A.K. (2018). Credit card fraud detection using adaboost and majority voting. IEEE Access, 6: 14277-14284. http://dx.doi.org/10.1109/ACCESS.2018.2806420

[4] Ileberi, E., Sun, Y., Wang, Z., (2022). A machine learning based credit card fraud detection using the GA algorithm for feature selection. Journal of Big Data, 9(24). http://dx.doi.org/10.1186/s40537-022-00573-8

[5] Alarfaj, F.K., Malik, I., Khan, H.U., Almusallam, N., Ramzan, M., Ahmed, M. (2022). Credit card fraud detection using state-of-the-art machine learning and deep learning algorithms. IEEE, 10: 39700-39715. http://dx.doi.org/10.1109/ACCESS.2022.3166891

[6] Melo-Acosta, G.E., Duitama-Muñoz, F., Arias-Londoño, J.D. (2017). Fraud detection in big data using supervised and semi-supervised learning techniques. IEEE Colombian Conference on Communications and Computing (COLCOM), pp. 1-6. http://dx.doi.org/10.1109/ColComCon.2017.8088206

[7] Goyal, R., Manjhvar, A.K. (2020). Review on credit card fraud detection using data mining classification techniques & machine learning algorithms. International Journal of Research and Analytical Reviews, 7(1): 972-975.

[8] Roy, A., Sun, J., Mahoney, R., Alonzi, L., Adams, S., Beling, P.(2018). Deep learning detecting fraud in credit card transactions. Systems and Information Engineering Design Symposium (SIEDS), pp. 129-134. http://dx.doi.org/10.1109/sieds.2018.8374722

[9] Wang, C., Wang, Y., Ye, Z., Yan, L., Cai, W., Pan, S. (2018). Credit card fraud detection based on whale algorithm optimized BP neural network. 13th International Conference on Computer Science & Education (ICCSE), pp. 1-4. http://dx.doi.org/10.1109/ICCSE.2018.8468855

[10] Chen, T., Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, pp. 785-794. https://doi.org/10.1145/2939672.2939785

[11] Oza-aditya, A. (2018). Fraud detection using machine learning. https://cs229.stanford.edu/proj2018/report/261.

[12] Saputra, A., Suharjito. (2019). Fraud detection using machine learning in e-commerce. International Journal of Advanced Computer Science and Applications(IJACSA), 10(9). http://dx.doi.org/10.14569/IJACSA.2019.0100943

[13] Madhav, V.V., Kumari, K.A. (2020), Analysis of credit card fraud data using PCA. IOSR Journal of Engineering (IOSRJEN), 10(1): 10-14.

[14] Maniraj, S.P., Saini, A., Ahmed, S., Sarkar, S.D. (2019). Credit card fraud detection using machine learning and

data science. International Journal of Engineering and Technical Research, 08(09). http://dx.doi.org/10.17577/IJERTV8IS090031

[15] Balogun, A.O., Basri, S., Abdulkadir, S.J., Hashim, A.S. (2019). Performance analysis of feature selection methods in software defect prediction: A search method approach. Appl. Sci., 9(13): 2764. http://dx.doi.org/10.3390/app9132764

[16] Benchaji, I., Douzi, S., Ouahidi, B.E. (2021). Credit card fraud detection model based on LSTM recurrent neural networks. Journal of Advances in Information Technology, 12(2): 113-118. http://dx.doi.org/10.12720/jait.12.2.113-118

[17] Zhang, Z., Chen, L., Liu, Q., Wang, P. (2020). A fraud detection method for low-frequency transaction. IEEE, 8: 25210-25220. http://dx.doi.org/10.1109/ACCESS.2020.2970614

[18] Pumsirirat, A., 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(ijacsa), 9(1): 18-25. http://dx.doi.org/10.14569/IJACSA.2018.090103

[19] Yang, C.F., Liu, G.J., Yan, C.G.(2020). A k-means-based and no-super-parametric improvement of AdaBoost and its application to transaction fraud detection. 2020 IEEE International Conference on Networking, Sensing and Control (ICNSC), pp. 1-5. http://dx.doi.org/10.1109/ICNSC48988.2020.9238121

[20] Zheng, L., Liu, G., Yan, C., Jiang, C., Zhou, M., Li, M. (2020). Improved TrAdaBoost and its application to transaction fraud detection. IEEE Transactions on Computational Social Systems, 7(5): 1304-1316. http://dx.doi.org/10.1109/TCSS.2020.3017013

[21] González Benaissa, A.A.R. (2021). IEEE-CIS fraud detection: A case study for fraudulent transaction detection based on supervised learning models. Revista Facultad de Ingenieria, https://bibliotecadigital.udea.edu.co/bitstream/10495/20175/10/AaronAlrachid_2021IEECISFraudPrediction

[22] Kim, J., Kim, H.J., Kim, H. (2019). Fraud detection for job placement using hierarchical clusters-based deep neural networks. Applied Intelligence, 49(8): 2842-2861. http://dx.doi.org/10.1007/s10489-019-01419-2

[23] Błaszczyński, J., de Almeida Filho, A.T., Matuszyk, A., Szeląg, M., Słowiński, R. (2021). Auto loan fraud detection using dominance-based rough set approach versus machine learning methods. Expert Systems with Applications, 163. http://dx.doi.org/10.1016/j.eswa.2020.113740

[24] Venkata Sailaja, N., Padma Sree, L., Mangathayaru, N. (2020). A new text categorisation strategy: Prototype design and experimental analysis. International Journal of Knowledge and Learning, 13(2): 146-167. http://dx.doi.org/ 10.1504/IJKL.2020.10028349

[25] Deris, M.M., Senan, N., Abdullah, Z., Mamat, R., Handaga, B. (2019). Dimensional reduction using conditional entropy for incomplete information systems. Parallel Computing Technologies, pp. 263-272. https://doi.org/10.1007/978-3-030-25636-4_21

[26] Zhu, D., Yan, C., Guang, M., Xie, Y. (2021). A novel information-entropy-based feature extraction method for transaction fraud detection. 2021 4th International Conference on Intelligent Autonomous Systems (ICoIAS), pp. 129-133. http://dx.doi.org/10.1109/ICoIAS53694.2021.00031

[27] Shaohui, D., Qiu, G., Mai, H., Yu, H. (2021). Customer transaction fraud detection using random forest. 2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE), pp. 144-147. http://dx.doi.org/10.1109/ICCECE51280.2021.9342259

[28] Lima, R.F., Pereira, A.C.M. (2017) Feature selection approaches to fraud detection in e-payment systems. In: Bridge, D., Stuckenschmidt, H. (eds) E-Commerce and Web Technologies. EC-Web 2016. Lecture Notes in Business Information Processing, 278: 111-126. http://dx.doi.org/10.1007/978-3-319-53676-7_9