International Journal of Multidisciplinary Engineering in Current Research - IJMEC
Volume 8, Issue 3, March-2023, http://ijmec.com/, ISSN: 2456-4265

CREDIT CARD FRAUD DETECTION USING ADABOOST ANDMAJORITY VOTING

Syed Ibrahim Amaan1, Syed Nazeer Uddin2, Syed Zaid Ahmed3, Dr.Suneel Pappala4
1,2,3 B.E. student, Department of IT, Lords Institute of Engineering and Technology, Hyderabad
4Associate Professor, Department of IT, Lords Institute of Engineering and Technology, Hyderabad
drsuneelpappala@lords.ac.in

ABSTRACT: Credit card fraud affects the financial services industry greatly. Credit card fraud costs American businesses and consumers billions of dollars annually. Due to privacy concerns, there is a dearth of studies that analyze actual credit card transactions. In this study, we use machine learning techniques to identify instances of credit card fraud. In the beginning, regular models are used. After that, we employ a combination of AdaBoost and majority voting techniques to create a hybrid approach. A publicly accessible data set of credit card transactions is utilized to assess the performance of the model. Then, a financial institution's actual credit card data is examined. The resilience of the algorithms is also evaluated by introducing noise into the data samples. The experimental findings show promise for the majority voting approach as a means of identifying credit card fraud with high rates of accuracy.

Keywords: AdaBoost; classification; credit card; fraud detection; predictive modelling; voting.

I. INTRODUCTION

The goal of fraud is financial or personal gain via illegal means [1]. There are two ways to protect yourself against financial loss caused by fraud: prevention and detection. Prevention of fraud is a preventative measure used to avoid the occurrence of fraud in the first place. However, when a scammer attempts a fraudulent transaction, fraud detection is essential.

The theft of credit card details for unauthorized transactions is what credit card fraud is all about. Physical and digital methods exist for processing credit card payments [2]. Credit card use is commonplace in real-world transactions. This may take place over the phone or online for digital transactions. The cardholder's information (number, expiration date, and verification code) is often sent through telephone or online.

Credit card use has skyrocketed over the last decade due to the proliferation of online shopping [3]. In 2011, over 320 million transactions were made in Malaysia using credit cards; by 2015, that figure had risen to almost 360 million. Credit card fraud has been on the rise with the popularity of using these cards. Credit card fraud continues to occur despite the widespread use of various authorisation methods. Internet fraud is especially common since identity and location may be concealed. The banking sector is feeling the effects of the increase in credit card theft. In 2015, worldwide credit card theft cost victims an estimated $21.84 billion [4].

When a business suffers a loss due to credit card theft, they are responsible for covering all related expenses [5, 6]. Since shops have to eat the loss themselves, prices may go up or promotions may be scaled down. Therefore,

it is crucial to lessen the loss, and a good fraud detection system is necessary to lessen the number of fraud occurrences. The identification of credit card fraud has been the subject of a number of research. Artificial neural networks, rule-induction approaches, decision trees, logistic regression, and support vector machines are all examples of popular machine learning and related methodologies [1]. These approaches may be employed alone, or they can be combined to generate hybrid models.

Here, we use a battery of twelve different machine learning algorithms to search for signs of credit card fraud. The algorithms cover the gamut from simple neural networks to sophisticated deep learning frameworks. They're tested using sample data and actual customer credit card information. Additionally, hybrid models are formed using AdaBoost and majority voting techniques. Noise is introduced into the real-world data set to further assess the models' resilience and dependability. The primary contribution of this work is an assessment of several machine learning models for credit card fraud detection using a real-world data set. This work uses a data set derived from the author's own three-months' worth of real credit card transactions, as opposed to the approaches utilized by other researchers using publicly accessible data sets.

II. RELATED STUDIES

Here we take a look at the pros and cons of using both stand-alone and combined machine learning algorithms in the financial sector. Credit card fraud, false financial statements, and other financial scams are only some of the topics covered.

A. SINGLE MODELS

In [6], the methods of Random Forest (RF), Support Vector Machine (SVM), and Logistic Regression (LOR) for detecting credit card fraud were analyzed. The dataset included purchases made in a single year. Algorithm performances were evaluated using data under-sampling, with RF showing superior results over SVM and LOR [6].

In [7], the use of an AIRS to identify credit card fraud was suggested. AIRS is superior to the original AIS model since it uses negative selection to boost accuracy. This led to a 25% boost in precision and a 40% decrease in system reaction time. [7].

In [8], the authors describe a method for detecting credit card fraud that uses a rule-based filter, a Dumpster-Shafer adder, a database of past transactions, and a Bayesian learner. Based on this initial view, the Dempster-Shafer theory then determined if the transaction in question was typical, suspicious, or out of the ordinary. In the event that a transaction was suspected, Bayesian learning was used to further assess the belief using transaction history [8]. The simulated success rate was 98%, according to the simulation [8]. Credit card fraud was identified using a modified Fisher Discriminant function in [9]. Because of this change, conventional functions are now far more attuned to the significance of special cases. The variances were calculated using a weighted average, which revealed lucrative trades. The improved function has proven profitable in simulations [9].

Credit card fraud case activity patterns are extracted using association rules in [10]. The Chilean retail sector was the primary focus of this data collection. Fuzzy Query 2+, a data mining program, was used to de-fuzzify and analyze the data samples [10]. By eliminating unnecessary restrictions, the output made the job of fraud analysts easier [10].

In [11], a method is offered to increase the efficiency with which credit card fraud situations are identified. A bank in Turkey provided the data set. The likelihood of fraud in each transaction was calculated. Using the Genetic Algorithm (GA) and scatter search, we were able to lower the number of incorrect classifications. When compared

to earlier findings [11], the new strategy yielded twice as much productivity.

The falsification of financial statements is another significant drain on resources. Financial statement fraud was detected using a range of techniques [12], such as support vector machines, linear ordinal regression, genetic programming, and probabilistic neural networks. We utilized a data collection with information on 202 Chinese businesses. Subsets of characteristics were chosen using the t-statistic; in two instances, 18 and 10 features were chosen, respectively. According to the findings, PNN was the most effective method, followed by GP [12]. Financial statement fraud was detected using Decision Trees (DT) and Bayesian Belief Networks (BNN) in [13]. Ratios extracted from the annual reports of 76 Greek industrial companies were used as input. The auditors confirmed the fraudulent nature of 38 financial statements. With an accuracy of 90.3%, BBN outperformed DT's 73.6% [13].

In order to identify accounting shenanigans, a computational fraud detection model (CFDM) was presented in [14]. Fraud was uncovered via analyzing textual information. Information was taken in the form of samples from 10-K reports filed with the SEC. The CDFM model was able to differentiate between legitimate and fraudulent submissions [14]. In [15], we see the proposal of a threshold-type detection approach for fraud detection based on the visual representation of user accounts. As a means of representation, the Self-Organizing Map (SOM) was used. Fraud in the areas of telecommunications, computer networks, and credit cards were examined using real-world data sets. By projecting high-dimensional data samples in a basic 2-dimensional space using the SOM [15], the findings were presented in a way that was visually appealing to both data analysts and non-experts.

The identification of fraud and the analysis of spending habits to identify fraudulent activities are discussed in [16]. It analyzed, filtered, and interpreted suspicious activities using the SOM. Hidden patterns in the input data were revealed via the application of clustering. Then, filters were utilized to cut down on overhead and speed up the process. The SOM was able to converge quickly because of the careful tuning of the number of neurons and the number of iteration steps. From what we could see, the final model was both effective and economical [16]. HYBRID SYSTEMS

Hybrid models are a synthesis of many separate models. In [17], the authors employ a model that combines the Multilayer Perceptron (MLP) neural network, the Support Vector Machine (SVM), the Logistic Regression (LOR), and the Harmony Search (HS) optimization to identify cases of corporate tax evasion. Finding optimal parameters for the classification models was facilitated by HS. The MLP with HS optimization achieved the greatest accuracy rates, 90.07% [17], when applied to data from Iran's food and textile industries. In [18], a hybrid clustering approach was used to identify fraud in lottery and online games by looking for outliers. In order to detect various forms of fraud, the system combined the results of online algorithms with statistical data taken from the input. Compressing the training data set into main memory allowed for progressive addition of fresh data samples to the stored data-cubes. The system's detection rate was 98%, and it only generated 0.1% false alarms [18].

Hybrid models based on clustering and classifier ensemble techniques were employed to address monetary hardship in [19]. For clustering, we utilized SOM and k-means, and for classification, we used LOR, MLP, and DT. Twenty-one hybrid models using various combinations of the aforementioned techniques were developed and tested on the dataset. The SOM coupled with the MLP classifier achieved the maximum level of performance and prediction accuracy [19]. In [20], a fraud detection model for business financial accounts was developed using the combination of many models, including RF, DR, Roush Set Theory (RST), and a back-propagation neural network. The data set was compiled from companies' financial statements covering the years 1998 through 2008. According to the findings, the best classification accuracy [20] was achieved by a model that included RF and RST.

Both [21] and [22] detail techniques for detecting motor insurance fraud. In [21], the authors offer a PCA-based RF model that uses the potential closest neighbor technique. Potential closest neighbor voting has replaced RF's customary majority vote. The experimental investigation used a total of 12 distinct data sets. Classification accuracy and variance were both improved in the PCA-based model compared to the RF and DT approaches [21]. For the purpose of detecting vehicle insurance fraud, [22] advocated using a genetic algorithm (GA) in conjunction with fuzzy c-means (FCM). Testing data was clustered and then classified as legitimate, malicious, or suspicious. After removing all legitimate and fraudulent records, DT, SVM, MLP, and the Group Method of Data Handling (GMDH) were applied to the remaining suspect instances. Both the specificity and sensitivity were maximized by the SVM [22].

III. MACHINE LEARNING ALGORITHMS

This experimental investigation employs a total of twelve algorithms. They work in tandem with AdaBoost and majority voting processes. The specifics are outlined below.

ALGORITHMS

Naive Bayes (NB) employs Bayes' theorem for classification under too optimistic assumptions of independence. It is considered that certain characteristics of a class are unrelated to others. In order to estimate the means and variances, just a little amount of training data is required.

Users benefit from having information presented in a tree structure since it's intuitive and easy to understand. The DT is a network of nodes used to make inferences about relationships between classes and individual attributes. Each node represents a possible feature-splitting rule. Up until the termination requirement is reached, new nodes are created. The designation for each class is dependent on how many samples are of that leaf type. While the Random Tree (RT) acts similarly to the DT operator, it only uses a random subset of characteristics for each split. It is able to learn from both nominal and numeric examples. A subset ratio parameter is used to determine the subset size.

An ensemble of random trees is generated using the Random Forest (RF) algorithm. The player determines how many trees are used. The final model uses a voting procedure applied to all of the generated trees to arrive at a consensus categorization. The Gradient Boosted Tree (GBT) is a set of models that may be used for either classification or regression. It makes use of forward-learning ensemble models, which provide forecasts by continuously refining their estimates. Boosting aids in raising the accuracy of the tree. A decision tree with a single branch is all that the Decision Stump (DS) can provide. It may be used to categorize unbalanced datasets. There are at least three types of nodes in an MLP network: input, hidden, and output. Except for the input nodes, all other nodes use a non-linear activation function. For instruction, it employs the supervised backpropagation method. The MLP variant employed in this research has automated training parameters including learning rate and hidden layer size adjustments. It employs a collection of networks that undergo simultaneous training with varying rates and quantities of hidden units.

The backpropagation technique is also used to train the Feed-Forward Neural Network (NN). Information only flows from the input nodes to the output nodes, via the hidden nodes, and the connections between the units do not form a directed cycle. When it comes to DL, a Multilayer Perceptron (MLP) network trained via stochastic gradient descent and backpropagation is at the heart of things. Neurons with tanh, rectifier, and maxout

activation functions are contained inside a vast number of hidden layers. Each node stores its own copy of the global model's parameters on its own data and regularly adds to the global model by means of model averaging. Scalar-variable relationships may be modeled using linear equation fitting in Linear Regression (LIR). Unknown model parameters are inferred from the data set and linear predictor functions are used to model the relationships. Model selection employs the Akaike criteria, a measure of the relative goodness of fit for a statistical model. Both nominal and numeric data are suitable for use in Logistic Regression (LOR). It calculates a predicted value for a binary outcome from a set of predictors.

Data for both classification and regression may be processed using the SVM. With SVM, a model is constructed by arbitrarily placing fresh samples into one of two categories, yielding a probabilistically invalid binary linear classifier. Data samples are shown as points in a space that has been mapped to provide as much separation as possible between samples belonging to various categories. A summaryof the strengths and limitations of the methods discussed earlier is given in Table I.

Table I: Strengths and Limitations of Machine Learning Methods

Model	**Strengths**	**Limitations**
Bayesian	Good for binary classification problems; efficient use of computational resources; suitable for real-time operations.	Need good understanding of typical and abnormal behaviors for different types of fraud cases
Trees	Easy to understand and implement; the procedures require a low computational power; suitable for real-time operations.	Potential of over-fitting if the training set does not represent the underlying domain information; re-training is required for new types of fraud cases.
Neural Network	Suitable for binary classification problems, and widely used for fraud detection.	Need a high computational power, unsuitable for real-time operations; re-training is required for new types of fraud cases.
Linear Regression	Provide optimal results when the relationship between independent and dependent variables are almost linear.	Sensitive to outliers and limited to numeric values only.
Logistic Regression	Easy to implement, and historically used for fraud detection.	Poor classification performances as compared with other data mining methods.
Support Vector Machine	Able to solve non-linear classification problems; require a low computational power; suitable for real-time operations.	Not easy to process the results due to transformation of the input data.

A. MAJORITY VOTING

Majority voting is frequently used in data classification, In every iteration t, the weak learner is chosen, and is allotteda coefficient, αt, so that the training error sum, Et, of the resulting t-stage boosted classifier is minimized, which involves a combined model with at least twoalgorithms. Each algorithm makes its own prediction for,

$$E_t = \mathbf{I}\,[F_{t-0}(x_i) + \alpha_t h(x_i)] \qquad (3)$$

every test sample. The final output is for the one that receives the majority of the votes, as follows.

Consider K target classes (or labels), with $C_i, \forall i \in \Lambda = \{1, 2, \dots, K\}$ represents the i-th target class predicted by a classifier. Given an input $\boldsymbol{x}$, each classifier provides a prediction with respect to the target class, yielding a total of$K$ prediction, i.e., $P_0, \dots, P_K$. Majority voting aims to producea combined prediction for input x, $(\mathbf{x}) = j$, $j \in \Lambda$ from all the K predictions, i.e., $p(\mathbf{x}) = j_k$, $k = 1, \dots, K$. A binary function can be used to represent the votes, *i.e.*, where $Ft–1(x)$ is the boosted classifier built in the previous stage, $E(F)$ is the error function, and $ft(x) = \alpha th(x)$ is weak learner taken into consideration for the final classifier.

AdaBoost tweaks weak learners in favor of misclassified data samples. It is, however, sensitive to noise and

outliers.As long as the classifier performance is not random, AdaBoost is able to improve the individual results from different algorithms.

IV. EXPERIMENTS

$_{k}$ $_{i}V$ (**x** ∈) = :1, if $p_k(\mathbf{x}) = i, i \in \Lambda 0$, otherwise (1)

In this section, the experimental setup is firstly detailed. Thisis followed by a benchmark evaluation using a publicly

Then, sum the votes from all *K* classifiers for each *Ci*, andthe label that receives the highest vote is the final (combined)
predicted class.

B. ADABOOST

Adaptive Boosting, often known as AdaBoost, is used to enhance the efficiency of various algorithms. The combined output of the boosted classifier, i.e., the accessible data set, is represented by a weighted sum of the individual outputs. Next, we do an analysis of the raw credit card data. RapidMiner Studio 7.6 has been used for all tests. All parameters in RapidMiner have been set to their default values. As CV may mitigate the bias introduced by random sampling during assessment, a 10-fold CV was used in the studies [23].

A. EXPERIMENTAL SETUP

In the credit card data set, the number of fraudulent

$$(x) = \mathbf{I}\, f_t(x) \qquad t\mathrm{L}0$$

transactions is usually a very small as compared with thetotal number of transactions. With a skewed data set, the in which each foot is a classifier (weak learner) that gives back the anticipated class given input x. For every input sample (xi), each weak learner predicts an outcome (h(xi)). This does not provide a true picture of how well the system functions. Misclassifying a genuine transaction as fraudulent results in subpar service to consumers, while failing to identify fraud instances results in financial loss for the business and its clients. The inefficiency of machine learning algorithms is directly linked to this issue of uneven data distribution. The findings tend to favor the sample type from which the bulk of data came. Bhattacharyya et al. [6], Duman et al. [24], and Phua et al. [25] have all employed under-sampling to address data-imbalance issues. In this study, the skewed data set is dealt with using under-sampling.

Although there is no one indicator that adequately describes both true and false positives and negatives, the Matthews Correlation Coefficient (MCC) [26] is a good overall metric. Taking into consideration both true and erroneous positives and negatives, MCC evaluates the quality of a two-class issue. It is a fair test, even when comparing groups of varying sizes. The MCC may be determined at a rate as low as 1%. Very little shifts can be seen in the MCC rates, with NB's MCC score increasing from 0.219 to 0.235.

Table III Results Of Adaboost

Model	Accuracy	Fraud	Non-fraud	MCC
NB	98.038%	82.520%	98.064%	0.235
DT	99.919%	81.098%	99.951%	0.775
RF	99.889%	42.683%	99.988%	0.604
GBT	99.903%	81.707%	99.935%	0.747
DS	99.906%	66.870%	99.963%	0.711
RT	99.866%	32.520%	99.982%	0.497
DL	99.915%	79.878%	99.950%	0.765
NN	99.933%	81.301%	99.965%	0.807
MLP	99.933%	80.894%	99.966%	0.806
LIR	99.907%	54.472%	99.985%	0.686
LOR	99.926%	79.065%	99.962%	0.786
SVM	99.927%	82.317%	99.957%	0.796

where the result of +1 indicates a perfect prediction, and −1 a total disagreement.

B. BENCHMARK DATA

Data is obtained from [27], which is open to the public. There was a total of 284,807 September 2013 transactions from European cardholders included. There is a large unbalance in the data set, which includes 492 fraudulent transactions. Because of the sensitive nature of the information, we give 28 transform-based primary components. The only information that has not been altered is the time and quantity information.

Table II displays the findings of many models. The rates of accuracy are high, hovering around 99%. However, the true result is more complicated than that since the rate at which fraud is detected ranges from 32.5 percent for RT to 83.3 percent for NB. In other words, non-fraud outcomes predominate accuracy rates in a way that is comparable to the accuracy rates. The best MCC score is generated by SVM (0.813), while the worst is generated by NB (0.219).

Table II: Results Of Various Individual Models

Model	Accuracy	Fraud	Non-fraud	MCC
NB	97.705%	83.130 %	97.730%	0.219
DT	99.919%	81.098 %	99.951%	0.775
RF	99.889%	42.683 %	99.988%	0.604
GBT	99.903%	81.098	99.936%	0.746

		%		
DS	99.906%	66.870 %	99.963%	0.711
RT	99.866%	32.520 %	99.982%	0.497
DL	99.924%	81.504 %	99.956%	0.787
NN	99.935%	82.317 %	99.966%	0.812
MLP	99.933%	80.894 %	99.966%	0.806
LIR	99.906%	54.065 %	99.985%	0.683
LOR	99.926%	79.065 %	99.962%	0.786
SVM	99.937%	79.878 %	99.972%	0.813

In addition to the standard models, AdaBoost has been used with all 12 models. The results are shown in Table III.It can be seen that the accuracy and non-fraud detection ratesare similar to those without AdaBoost. However, the fraud detection rates increase from 79.8% to 82.3% for SVM. Some models suffer a minor reduction in the fraud detection.

Based on the models that produce good rates in Table II, the majority voting method is applied to the models. A totalof 7 models are reported in Table IV. The accuracy rates areall above 99%, with DS+GBT yields a perfect non-fraud rate.The best fraud detection rate is achieved by NN+NB at 78.8%. The highest MCC score at 0.823 is yielded by NN+NB, which is higher than those form individual models.

Table Iv: Results Of Majority Voting

Model	**Accuracy**	**Fraud**	**Non-fraud**	**MCC**
DS+GBT	99.848%	11.992 %	100.000%	0.343
DT+DS	99.850%	14.024 %	99.998%	0.361
DT+GBT	99.920%	60.366 %	99.988%	0.737
DT+NB	99.932%	72.967 %	99.978%	0.788
NB+GBT	99.919%	66.463 %	99.976%	0.742

NN+NB	99.941%	78.862 %	99.978%	0.823
RF+GBT	99.865%	23.780 %	99.996%	0.468

Saia and Carta's findings, which analyzed the identical data set using a 10-fold CV assessment, are utilized for comparison. The results are shown in Table V below. Frequency-Domain (FD) and Random Forest (RF) models were employed in this study. The sensitivity rate, as described in, is the same as the non-fraud detection rate in Tables II–IV; it is the percentage of transactions that are accurately categorized as legal. As may be shown in Table V, RF can attain a maximum of 95% accuracy and 91% sensitivity. Most of the individual models tested in this research achieve accuracy and non-fraud (sensitivity) rates of above 99%.

Table V: Performance Comparison With Results Extracted From [28]

Model	**Accuracy**	**Sensitivity**
FD	77%	76%
RF	95%	91%

REAL-WORLD DATA

The experiment makes use of a genuine data collection including credit card information from a Malaysian bank. The data was collected from cardholders throughout South and Southeast Asia in the months of February and April of 2017. There are a total of 287,224 transactions documented, including 102 suspected instances of fraud. A sequence of purchases over time makes up the data. No personally identifiable information is included in order to protect the privacy of our customers. Table VI lists the characteristics that were included in the experiment.

Table Vi: Features In Credit Card Data

Code	**Description**
DE002	Primary account number (PAN)
DE004	Amount, transaction
DE006	Amount, cardholder billing
DE011	System trace audit number
DE012	Time, local transaction
DE013	Date, local transaction
DE018	Merchant type
DE022	Point of service entry mode
DE038	Authorization identification response
DE049	Currency code, transaction (ISO 4217)
DE051	Currency code, cardholder billing (ISO 4217)

Eleven distinct aspects are used. All except the final two codes are derived from ISO 8583 [29], whereas those two are from ISO 4217. To prevent identity theft, the actual 16-digit credit card number in a PAN is disguised by a continuously generated series of digits. Table VII displays the outcomes of many different models. Except for support vector machines (SVM), all accuracy ratings are in excess of 99%. Except for support vector machines, NB, DT, and LIR all have flawless non-fraud detection rates. The MCC rates offered by NB, DT, RF, and DS are the best available (0.990). From 7.4 percent for LIR to one hundred percent for RF, GBT, DS, NN, MLP, and LOR, the fraud detection rates are all over the map.

Table Vii: Results Of Various Individual Models

Model	**Accuracy**	**Fraud**	**Non-fraud**	**MCC**
NB	99.999%	98.039%	100.000%	0.990
DT	99.999%	98.039%	100.000%	0.990
RF	99.999%	100.000%	99.999%	0.990
GBT	99.999%	100.000%	99.999%	0.986
DS	99.999%	100.000%	99.999%	0.990
RT	99.992%	80.392%	99.999%	0.886
DL	99.985%	93.137%	99.987%	0.819
NN	99.997%	100.000%	99.997%	0.963
MLP	99.997%	100.000%	99.997%	0.954
LIR	99.965%	7.407%	100.000%	0.272
LOR	99.999%	100.000%	99.999%	0.981
SVM	95.564%	9.804%	95.595%	0.005

All individual models have been trained using AdaBoost, same as in the benchmark experiment. You may see the outcomes in Table VIII. Similar rates of accuracy and non-fraud detection are shown when AdaBoost is used as when it is not. AdaBoost increases the accuracy of fraud detection, most notably for NB, DT, and RT, which achieve 100% precision. When compared to other methods, LIR's jump in accuracy from 7.4% to 94.1% is by far the most impressive. The effectiveness of AdaBoost in boosting the efficiency of separate classifiers is shown here. NB and RF have the best MCC scores, both of which are 1.

Table Viii Results Of Adaboost

Model	Accuracy	Fraud	Non-fraud	MCC
NB	100.000%	100.000%	100.000%	1.000
DT	99.999%	100.000%	99.999%	0.990
RF	100.000%	100.000%	100.000%	1.000
GBT	99.999%	100.000%	99.999%	0.986
DS	99.999%	100.000%	99.999%	0.990
RT	100.000%	100.000%	100.000%	0.995
DL	99.994%	96.078%	99.995%	0.917
NN	99.998%	100.000%	99.998%	0.967
MLP	99.996%	100.000%	99.996%	0.950
LIR	99.992%	94.118%	99.994%	0.890
LOR	99.999%	100.000%	99.999%	0.981
SVM	99.959%	1.961%	99.994%	0.044

The identical models from the reference experiment are then subjected to the majority voting procedure. Table IX displays the findings. There is either one hundred percent accuracy or near one hundred percent non-fraud detection. The detection rate for fraud is 100% for DS+GBT, DT+DS, DT+GBT, and RF+GBT. When comparing MCC scores, they are all quite near to 1, if not exactly there. Voting as a group produces more accurate outcomes than any one model could on its own.

Table Ix: Results Of Majority Voting

Model	Accuracy	Fraud	Non-fraud	MCC
DS+GBT	100.000%	100.000%	100.000%	0.995
DT+DS	100.000%	100.000%	100.000%	0.995
DT+GBT	100.000%	100.000%	100.000%	1.000

DT+NB	99.999%	98.039%	100.000%	0.990
NB+GBT	99.999%	98.039%	100.000%	0.990
NN+NB	99.998%	95.098%	100.000%	0.975
RF+GBT	99.999%	100.000 %	99.999%	0.990

All real-world data samples are polluted with noise at 10%, 20%, and 30% to further assess the resilience of the machine learning methods. All data features have noise applied to them. The MCC score is shown in Figure 2, while the fraud detection rate is shown in Figure 1. As predicted, the detection rate for fraud and the MCC rate both drop when noise is introduced. In terms of accuracy and MCC, majority vote of DT+NB and NB+GBT fares the lowest. Even with 30% noise in the data set, the accuracy rates of DS+GBT, DT+DS, and DT+GBT remain over 90%.

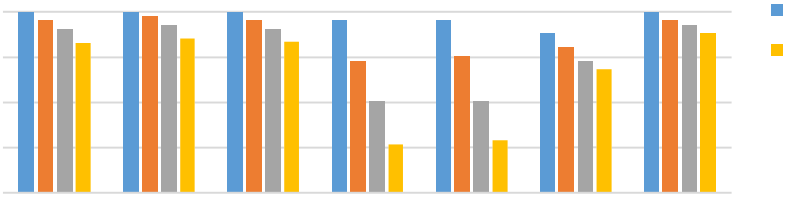

FIGURE 1. Fraud detection rates with different percentages of noise

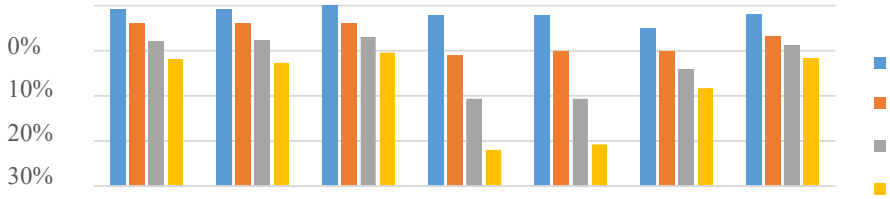

FIGURE 2. MCC scores with different percentages of noise

V. CONCLUSIONS

In this work, we report the results of our research into the use of machine learning algorithms for the purpose of detecting credit card fraud. Several commonplace models, such as NB, SVM, and DL, have been put through an empirical test. Individual (typical) models and hybrid models employing AdaBoost and majority voting combination techniques have been evaluated using a publicly accessible credit card data set. Since the MCC metric considers both correct and incorrect predictions, it has been widely used as a performance statistic. Using a simple majority, the highest possible MCC score is 0.823. Evaluators have also made use of a real-world credit card data set provided by a banking firm. Both standalone and combined models have been used. Using AdaBoost and majority vote approaches, we were able to get an MCC score of 1. Noise of 10%-

30% was introduced into the data samples to better analyze the hybrid models. With 30% noise introduced to the data set, the best MCC score was 0.942, achieved by majority vote. This demonstrates that the majority voting technique maintains its performance stability even while dealing with noise.

The methodologies explored in this study will be used to the development of online instructional frameworks in forthcoming research. Further, we want to look at other types of online education. In the future, online learning may even allow for real-time fraud detection. In turn, this will aid in the detection and prevention of fraudulent transactions before they occur, hence reducing daily losses in the banking industry.

REFERENCES

[1] Y. Sahin, S. Bulkan, and E. Duman, "A cost-sensitive decision tree approach for fraud detection," *Expert Systems with Applications*, vol.40, no. 15, pp. 5916–5923, 2013.

[2] A. O. Adewumi and A. A. Akinyelu, "A survey of machine-learning and nature-inspired based credit card fraud detection techniques," *International Journal of System Assurance Engineering and Management*, vol. 8, pp. 937–953, 2017.

[3] A. Srivastava, A. Kundu, S. Sural, A. Majumdar, "Credit card fraud detection using hidden Markov model," *IEEE Transactions on Dependable and Secure Computing*, vol. 5, no. 1, pp. 37–48, 2008.

[4] The Nilson Report (October 2016) [Online]. Available: https://www.nilsonreport.com/upload/content_promo/The_Nilson_R eport_10-17-2016.pdf

[5] J. T. Quah, and M. Sriganesh, "Real-time credit card fraud detection using computational intelligence," *Expert Systems with Applications*, vol. 35, no. 4, pp. 1721–1732, 2008.

[6] S. Bhattacharyya, S. Jha, K. Tharakunnel, and J. C., "Data mining forcredit card fraud: A comparative study," *Decision Support Systems*, vol. 50, no. 3, pp. 602–613, 2011.

[7] N. S. Halvaiee and M. K. Akbari, "A novel model for credit card fraud detection using Artificial Immune Systems," *Applied Soft Computing*, vol. 24, pp. 40–49, 2014.

[8] S. Panigrahi, A. Kundu, S. Sural, and A. K. Majumdar, "Credit card fraud detection: A fusion approach using Dempster–Shafer theory andBayesian learning," *Information Fusion*, vol. 10, no. 4, pp. 354–363, 2009.

[9] N. Mahmoudi and E. Duman, "Detecting credit card fraud by modified Fisher discriminant analysis," *Expert Systems with Applications*, vol. 42, no. 5, pp. 2510–2516, 2015.

[10] D. Sánchez, M. A. Vila, L. Cerda, and J. M. Serrano, "Association rules applied to credit card fraud detection," *Expert Systems with Applications*, vol. 36, no. 2, pp. 3630–3640, 2009.

[11] E. Duman and M. H. Ozcelik, "Detecting credit card fraud by geneticalgorithm and scatter search," *Expert Systems with Applications*, vol. 38, no. 10, pp. 13057–13063, 2011.

[12] P. Ravisankar, V. Ravi, G. R. Rao, and I. Bose, "Detection of financial statement fraud and feature selection using data mining
techniques," *Decision Support Systems*, vol. 50, no. 2, pp. 491–500, 2011.

[13] E. Kirkos, C. Spathis, and Y. Manolopoulos, "Data mining techniques for the detection of fraudulent financial statements," *Expert Systems with Applications*, vol. 32, no. 4, pp. 995–1003, 2007.

[14] F. H. Glancy and S. B. Yadav, "A computational model for financial reporting fraud detection," *Decision Support Systems*, vol. 50, no. 3, pp. 595–601, 2011.

[15] D. Olszewski, "Fraud detection using self-organizing map visualizingthe user profiles," *Knowledge-Based Systems*, vol. 70, pp. 324–334, 2014.

[16] J. T. Quah and M. Sriganesh, "Real-time credit card fraud detection using computational intelligence," *Expert Systems with Applications*, vol. 35, no. 4, pp. 1721–1732, 2008.

[17] E. Rahimikia, S. Mohammadi, T. Rahmani, and M. Ghazanfari, "Detecting corporate tax evasion using a hybrid intelligent system: Acase study of Iran," *International Journal of Accounting Information Systems*, vol. 25, pp. 1–17, 2017.

[18] I. T. Christou, M. Bakopoulos, T. Dimitriou, E. Amolochitis, S.Tsekeridou, and C. Dimitriadis, "Detecting fraud in online games of chance and lotteries," *Expert Systems with Applications*, vol. 38, no. 10, pp. 13158–13169, 2011.

[19] C. F. Tsai, "Combining cluster analysis with classifier ensembles to predict financial distress" *Information Fusion*, vol. 16, pp. 46–58,2014.

[20] F. H. Chen, D. J. Chi, and J. Y. Zhu, "Application of Random Forest,Rough Set Theory, Decision Tree and Neural Network to Detect Financial Statement Fraud–Taking Corporate Governance into Consideration," In *International Conference on Intelligent Computing*, pp. 221–234, Springer, 2014. Y. Li, C. Yan, W. Liu, and M. Li, "A principle component analysis- based random forest with the potential nearest neighbor method for automobile insurance fraud identification," *Applied Soft Computing*, to be published. DOI: 10.1016/j.asoc.2017.07.027.

[21] S. Subudhi and S. Panigrahi, "Use of optimized Fuzzy C-Means clustering and supervised classifiers for automobile insurance fraud detection," *Journal of King Saud University-Computer and Information Sciences*, to be published. DOI: 10.1016/j.jksuci.2017.09.010.

[22] M. Seera, C. P. Lim, K. S. Tan, and W. S. Liew, "Classification of transcranial Doppler signals using individual and ensemble recurrent neural networks," *Neurocomputing*, vol. 249, pp. 337-344, 2017.

[23] E. Duman, A. Buyukkaya, and I. Elikucuk, "A novel and successful credit card fraud detection system Implemented in a Turkish Bank," In *IEEE 13th International Conference on Data Mining Workshops (ICDMW)*, pp. 162–171, 2013.

[24] C. Phua, K. Smith-Miles, V. Lee, and R. Gayler, "Resilient identity crime detection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 3, pp. 533–546, 2012.

[25] M. W. Powers, "Evaluation: From Precision, Recall and F-Measure toROC, Informedness, Markedness & Correlation" *Journal of MachineLearning Technologies*, vol. 2, no. 1, pp. 37–63, 2011.

[26] Credit Card Fraud Detection [Online]. Available: https://www.kaggle.com/dalpozz/creditcardfraud

[27] R. Saia and S. Carta, "Evaluating Credit Card Transactions in the Frequency Domain for a Proactive Fraud Detection Approach," In *Proceedings of the 14th International Joint Conference on e- Business and Telecommunications*, vol. 4, pp. 335–342, 2017.

[28] ISO 8583-1:2003 Financial transaction card originated messages [Online]. Available: https://www.iso.org/standard/31628.html