

Credit Card Fraud Detection

Viikrant Tomar

Information Technology
Rajkiya Engineering College
Ambedkar Nagar

Vikrantbpt839@gmail.com

Sarvesh Kumar Maury

Information Technology
Rajkiya Engineering College
Ambedkar Nagar

maurysarvesh221296@gmail.com

Dr.Ramesh Chand Pandey

Assistant professor
Department of Information Technology
Rajkiya Engineering College
Ambedkar Nagar

Abstract

Advances in communication technology and e-commerce have made the credit card a popular online payment method. Therefore, security in this system is highly expected to prevent fraud. Credit card transaction fraud transactions are increasing year by year. In this way, researchers also experiment with novels to detect and prevent such fraud. However, there is always a need for some strategy to master this trick. This paper proposes a system to detect fraud in credit card details using an unregulated neural learning process (NN). The proposed method exceeds the existing Auto Encoder (AE), Local Outlier Factor (LOF), Isolate Forest (IF) and K-Means clustering methods. The proposed NN fraud detection method works with 99.87% accuracy and the existing AE, IF, LOF and K Means methods provide 97 %, 98 %, 98 % and 99.75 % accuracy.

Keywords

Unattended Reading, Unwanted Detection, Fraud Detection, Automated Encoder, Credit Card

I. INTRODUCTION

Fraud is a criminal fraud intended to bring financial or personal gain [1]. To escape fraud, two methods can be used: fraud prevention and fraud detection. Fraud prevention is a good way, where it stops fraud from being original. alternatively, fraud detection is required when fraudulent transactions are attempted by the fraudster. Credit card fraud is related to the illegal use of credit card information for any purpose. Credit card transactions can be physical or digital [2]. In a physical transaction, a credit card is physically available at the time of the transaction. In digital marketing, this happens over the phone or online. Cardholders in particular know their card number, expiration date, and other important information by telephone or website. in the last few years in the rise of commerce, the consumer of credit cards has risen [3]. The number of credit card transactions in 2011 in Malaysia was almost 320 million, and increased in 2015 to 360 million. with the user, credit card fraud is also increasing rapidly. While the code of conduct is being used, credit card fraud cases have not been successful. A fraudster loves the Internet as his or her identity and hides his or her location. Credit card fraud has a huge impact on the financial industry. Global credit card fraud in 2015 reached USD \$ 21.84 billion

[4]. Credit card loss affects retailers, where they bear all costs, including card issuers, costs, and administration costs [5]. As retailers need to lose, some goods are more expensive, or discounts and incentives are reduced. Therefore, it is important to minimize losses, and an effective fraud detection system to reduce or eliminate fraud cases is essential. There have been various studies on finding credit card fraud. Mechanical learning methods and related methods are widely used, including artificial neural networks, law enforcement techniques, decision trees, retransmission, and vector support systems [1]. These methods are used independently or by combining several methods together to form hybrid models. In this paper, a total of twelve machine learning algorithms are used to detect credit card fraud. Algorithms range from standard NN (Neural networks) to deep learning models. They are tested using both international credit card data sets. In addition, the AdaBoost methods and multiple voting methods are used to build hybrid models. Continuing to test the durability and reliability of the models, sound can be heard in the actual data set. An important contribution of this paper is the analysis of the various types of machine learning with real-world credit card data reserved for fraud detection. While some researchers have used different methods for publicly available data sets, the data set used in this paper is extracted from credit card transaction information for more than three months. The editing of this paper is as follows. In Phase II, courses related to the same machine learning and hybrid technology for financial applications are offered. The machine learning methods used in this study are presented in section III. A standardized and actual credit card data settings are presented in section IV. Concluding remarks and recommendations for ongoing work are provided in section V.

II. RELATED STUDIES

In this article, one-to-one and hybrid machine learning algorithms for financial applications are updated. Various financial claims from credit card fraud to fraudulent financial statements are being reviewed.

A. SPECIAL MODELS

With the discovery of credit card fraud, Random Forest (RF), Support Vector Machine, (SVM) and Logistic Regression (LOR) were reviewed in [6]. The data set contains a one-year transaction. Sub-sample data was used to update the algorithm representation, with RF showing better results as

compared to SVM and LOR [6]. The Artificial Immune Recognition System (AIRS) for detecting credit card fraud was prioritized in [7]. AIRS is an improvement over the standard AIS model, where a poor collection has been used to obtain high accuracy. This increases accuracy by 25% and reduces system response time by 40% [7]. The credit card fraud detection model was prioritized in [8], consisting of a law-based filter, a Dumpster - Shafer adder, a transaction histories database, and a Basezi student. Dempster's view - Shafer combines the many details of the evidence and presents the original belief, which is used to classify the contract agreement as standard, questionable, or uncommon. when the transaction was removed it was doubtful, the belief was re-examined using the transaction history from the Basesi study [8]. the results show a 98% true positive rate [8]. The modified Fisher Discriminant function has been used to detect credit card fraud in [9]. Changes have made traditional activities more efficient in important situations. The weighted measure was self-calculating for a different calculation, which allows for the study of profitable transactions. The results from the modified work confirm that it can generate more benefits [9]. Organizational rules are available by issuing codes of conduct on credit card fraud cases to [10]. The data is set to interest in retail companies in Chile. Data analysis was performed with DE fuzzified data and analyzed using the Sensitive Data Mining 2+ tool [10]. The result has reduced the number of unequal rules, which clarify the work of fraudulent analysts [10]. To improve the detection of credit card fraud cases, a solution was put in [11]. Data stored in the Turkish bank was used. Each transaction is rated as fraudulent or otherwise. Error in reduced calculation levels using Genetic Algorithm (GA) and search spreads. The prioritization approach is twice as effective, compared with previous results [11]. Other major financial losses are linked to the fraud of the financial statements. Several strategies including SVM, LOR, Genetic Programming (GP) and Probabilistic Neural Network (PNN) have been used to detect fraud in the financial statements [12]. A data set involving 202 Chinese companies was canceled. The t number was issued for a subset of a set of factors, in which 18 and 10 factors were collected in two cases. The results showed that PNN did the best, followed by GP [12]. Decision Trees (DT) and Bayesian Belief Networks (BBN) were dismissed [13] for the purpose of detecting fraudulent financial statements. The input contains estimates taken from the financial statements of 76 Greek manufacturing firms. The 38 financial statements have been confirmed as fraudulent by the auditors. BBN received the best accuracy of 90.3%, while DT obtained 73.6% [13]. A computational fraud detection model (CFDM) was introduced at [14] to detect financial reporting fraud. It automatically detects text error detection. Data trials from 10-K completions in the Security and Exchange Commission were used. The CDFM model has been able to distinguish counterfeit forms from non-fraudulent forms [14]. The method of detecting fraud based on the appearance of user accounts and the detection of the threshold type was introduced in [15]. The Self-Determination Map (SOM) was released as a visual aid. The original data sets were linked to telecommunication fraud, computer network infiltration, and credit card fraud. The results were demonstrated with a significant draw on data analysts and non-experts, as high-resolution data tests were performed in a simple 2-

dimensional environment using SOM [15]. The discovery of fraud and understanding of ways to spend money to identify potential fraud cases was described in [16]. Use SOM to interpret, filter, and evaluate fraudulent methods. The combination was removed to identify hidden patterns in the input data. After that, filters were removed to reduce the cost and processing time. By setting the appropriate numbers of neurons and iteration steps, SOM was able to quickly assemble. The emerging model has proven to be an effective and inexpensive method [16]

B. HYBRID MODELS

Hybrid models are organized into multiple individual models. The hybrid model containing the Multilayer Perceptron (MLP) network of neural network, SVM, LOR, and Harmony Search (HS) performance was removed in [17] to detect tax evasion in companies. HS was instrumental in obtaining the best parameters for class models. Using data from the food and textiles sector of Iran, MLP with HS optimization obtained a maximum accuracy of 90.07% [17]. A hybrid integration system with external discovery capabilities was released on [18] to detect fraud in online games and games. The system is composed of online algorithms with statistical data from input data to identify multiple types of fraud. The training data set is compressed into large memory while new data tracks can be added further to the cube (s) of stored data. The system detected a high detection rate of 98%, with an alarming 0.1% false alarm rate [18]. To address financial distress, integration and segregation methods were developed to make hybrid models in [19]. The SOM and k-means algorithms are distributed for integration, while LOR, MLP, and DT are distributed for integration. Based on these methods, a total of 21 hybrid models with different combinations were created and calculated by data set. SOM with MLP fragmentation did the best, providing the highest accuracy of prediction [19]. The integration of multiple models, namely RF, DR, Roush Set Theory (RST), and neural-back-propagation network was eliminated [20] to create a model for detecting fraudulent business financial statements. The company's financial statements for 1998 to 2008 were released as set as data. The results showed that the hybrid model of RF and RST provided the highest accuracy of the categories [20]. Methods for detecting car insurance fraud are presented in [21] and [22]. The main RF (PCA) - based (PCA) RF model combined with the nearest neighbor was set to [21]. The traditional majority vote on the RF has been replaced by the nearest neighboring route. A total of 12 different data sets were distributed in the experimental study. The PCA-based model resulted in higher separation accuracy and lower variability, compared to those from RF and DT methods [21]. The sophisticated c-means GA (FCM) was introduced at [22] to detect car insurance fraud. Test records are categorized as real, dangerous or suspicious categories based on structured collections. By rejecting real and fraudulent records, suspicious cases continue to be investigated using DT, SVM, MLP, and Group Method of Data Handling (GMDH). SVM produced higher specificity and sensitivity than weight [22].

III. MACHINE LEARNING ALGORITHMS

Twelve algorithms are used here for testing and reading. They are used in accordance with AdaBoost and many voting methods. Details are provided below.

A. ALGORITHMS

The Naïve Bayes (NB) use the Bayes' theory on a solid or naïve independent basis for planning. Certain aspects of a category are considered to be incompatible with others. It requires a minimum set of data to guess the methods and variations needed to differentiate. The introduction of data into the structure of the tree structure helps to easily define users. Decision Tree (DT) is a set of nodes that decides on features linked to specific classes. Every node represents a separate rule for review. New locations are established until the stopping process is met. The class label is determined by the number of trials in a particular leaf. Random Tree (RT) is set as the DT operator, except that in all divisions, only the lower set for updates is available. Learns from cost and price data trends. Subset size is defined using the subset rate parameter. Random Forest (RF) forms a cluster of random trees. User sets the number of trees. The emerging model uses the polling of all created trees to assess the effect of the final split. Gradient Boosted Tree (GBT) is a group of stages of planning or retreat. It uses advanced study group models, which obtain predictive results using progressively measuring estimates. Encouragement helps to improve the accuracy of the tree. Decision Stump (DS) creates a decision tree with just one split.

networks trained in the same complexity and number of hidden units. Feed-Forward Neural Network (NN) uses a retrospective algorithm to train with. Communication between units does not miss the target cycle, and data only progresses from input nodes to output nodes, through encrypted nodes. Deep Learning (DL) is based on a MLP network trained using stochastic gradient collapse with backpropagation. It contains a large number of hidden layers containing neurons with tanh, rectifier, and high activation functions. Each node records a copy of the global model parameters in local data, and contributes periodically to the global model using the model scale. Linear Regression (LIR) modulates the relationship between scalar variables by inserting direct activity into visual data. Relationships are simulated using an active line prediction, with unspecified model parameters measured in a set of data. The Akaike rating, the average equity rating for a mathematical model, is deducted from model selection. Logistic Regression (LOR) can manage data for both composing and number features. It measures the probability of a binary response based on one or more forecasts. SVM can deal with split and reverse data. SVM builds a model by offering new tests in one category or another, creating uninterrupted static binary separation. Represents data tests such as points in a map-created space so that data tests for different categories can be categorized as broadly as possible. A summary of the strengths and limitations of the methods discussed earlier is provided in Table I.

TABLE I

Model	Strengths	Limitations
Bayesian	It is good to distinguish a binary problem; efficient use of computer resources; a good suit for real-time performance.	Require a good understanding of the common and uncommon act of various types of fraud cases.
Trees	Understanding and implementing is easy, the lower computer power required in the process, the better the real-time performance.	It also requires training when the training set does not reflect the nature of the basic data and causes the power of excessive equilibrium.
Neural Network	It is suitable for binary split problems, and has been misused for fraudulent detection.	High computer power required, not suitable for real-time operation, If a new type of fraud occurs that requires retraining.
Linear Regression	Gives good results in a line of independent and dependent variations.	It is very sensitive to the actual numbers only.
Logistic Regression	Implementation is easy, and is often used to detect fraud.	When it comes to comparisons with other algorithms in the data mine, their performance is much worse.
Support Vector Machine	It can solve the problem of offline separation easily, low computer power is required, very suitable for real-time operation.	Due to the input of input data, it is difficult to process the result.

It can be spread to separate unequal data sets. The MLP network contains at least three layers of layers, namely, input, hidden, and output. Every node uses a non-line activation function, and input nodes deviation. It uses a back-to-back distribution algorithm for training. The MLP cast-off version in this study is able to adjust the reading level and layer size automatically hidden during training. It uses a group of

B. MAJORITY VOTING

Voting for the majority is often distributed through data segregation, requiring an integrated model with at least two algorithms. Each algorithm makes its

prediction for each test test. The final result is the one who gets the most votes, next.

Consider the K (or labels) classes of K , with $c_i, i \in \Lambda = \{1, 2, \dots, k\}$ representing the predicted i^{th} category in relation to the target category, giving the K predictive context p_1, \dots, p_k . The majority vote aims to make a combined prediction of input x , $p(x) = j, j \in \Lambda$ in all K predictions, i.e., $p_k(x) = j_k, k = 1, \dots, k$. Binary activity can be removed to represent votes, e.g.

$$\forall k ((x \in c_i) = \begin{cases} 1 & \text{if } p_k(x) = i, i \in \Lambda @ 0, \\ 0 & \text{otherwise} \end{cases})$$

After that, summarize the votes from all K dividers for all C_i , and the label that gets the highest number of votes in the last category (included)

C. ADABOOST

Adaptive Boosting or AdaBoost is distributed in conjunction with different types of algorithms to improve their performance. The results are compiled using a weighted sum, which represents the combined effects of an enlarged divider, i.e.,

$$F_T(x) = \sum_{t=1}^T \alpha_t f_t(x) \quad (2)$$

where each f_t divider (weak reader) returns the predicted category in relation to input x . Each weak student gives an exclusion prediction, $h(x_i)$, throughout the training test.

In each iteration t , the weaker student is selected, and given a coefficient, α_t , so that the error of the training error, E_t , of the increased t -stage of the reduced division,

$$E_t = \sum_i E [F_{t-1}(x_i) + \alpha_t h(x_i)] \quad (3)$$

where $F_{t-1}(x)$ is an advanced separator formed in the final phase, $E(F)$ is a function of error, and $f_t(x) = \alpha^{\text{th}}(x)$ is a weak reader considered for the final division. AdaBoost converts weak students to favor anonymous data testing. However, it is sensitive to noise and other areas. As long as the partition performance is not random, AdaBoost is able to improve individual results from different algorithms.

IV. EXPERIMENTS

At this stage, the initial test setup is detailed. This is followed by a benchmark test using a publicly available data set. Real-world credit card data is being tested. All tests were monitored using RipidMiner Studio 7.6. Standard settings for all parameters in RapidMiner deleted. A 10-fold cross-validation (CV) has been removed from the test as it may reduce the bias associated with randomized testing in the test phase [23].

A. TEST SCHEDULE

In a credit card data set, the number of fraudulent transactions is usually very small compared to the total transaction amount. With a twisted data set, the accuracy of the result does not reflect an accurate representation of system performance. Misappropriation of legal transactions results in poor customer service, and failure to obtain fraud claims results in loss to the financial institution and customers. This data diversity problem causes operational issues in machine learning algorithms. The multi-test category contributes to the results. The sample below was costly by Bhattacharyya et al. [6], Duman et al. [24], and Phua et al. [25] addressing data diversity problems. Therefore, less sampling is more expensive in this paper to handle preset data. While there is no better way to report good and false and bad using one index, the most common scales are the Matthews Correlation Coefficient (MCC) [26]. The MCC measures the quality of a two-class problem, which draws true and false benefits. Average scale, even if classes come in different sizes. MCC can be calculated using:

$$MCC = x = (TP * TN - FP * FN) / \sqrt{((TP + FP)(TP + FN)(TN + FP)(TN + FN))} \quad (4)$$

when the +1 result indicates a positive prediction, and -1 is totally inconsistent.

B. BENCHMARK DATA

A set of publicly available data is downloaded from [27]. It has a total of 284,807 transactions made in September 2013 by European policyholder(cardholders). The data set contains 492 fraudulent transactions, which are very different. Due to a privacy issue, a total of 28 key elements based on the change have been set. Only time and value data can be converted, and set as such. Results from various models are shown in Table II. It can be seen that the accuracy of the weight is high, the frequency is around 99%. This however is not a real result as the weight of fraud detection varies from 32.5% of RT to 83% of NB. The weight of a non-fraud acquisition is the same as the weight of accuracy, i.e., non-fraudulent results govern the accuracy levels. SVM made the highest MCC score of 0.813, while the lowest from NB had a MCC note of 0.219.

In addition to standard models, AdaBoost has been released in all twelve versions. The results are shown in Table III. It can be seen that the availability of precision and non-fraud has the same weight as those without AdaBoost. However, the fraud detection measured an increase from 79.8% to 82.3% of SVM. Some models suffer from a slight decrease in detection of fraud.

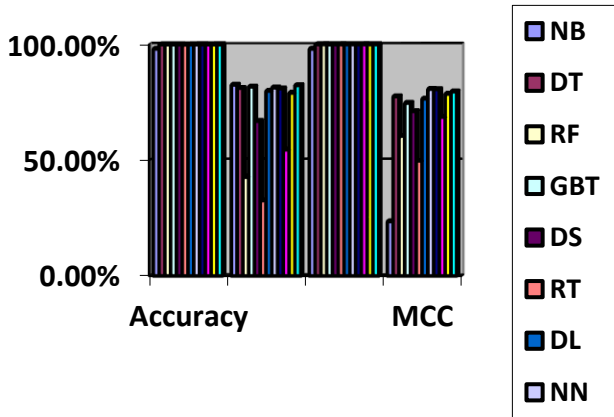
TABLE II
RESULTS OF VARIOUS INDIVIDUAL MODELS

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

average up to 1%. Weighted MCC shows very small changes, with NB being able to improve its MCC points from 0.219 to 0.235.

TABLE III
OUTCOMES OF ADABOOST



According to the models that produce good quality weights in Table II, the most voting method is included in the models. 7 models are announced in Table IV. The weight of the total weight is more than 99%, with DS + GBT producing total weight which is not fake. The weight for getting the best fraud is found by NN + NB at 78.8%. The high MCC rating at 0.823 was reversed by NN + NB, which is higher than that of 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

By comparing performance, the results presented in Saia and Carta [28] were discarded, using the same data set in a 10-fold CV test. Results are shown in Table V. Two models are extracted from [28], one from Frequency Domain (FD) and the other from Random Forest (RF). Sensitivity is weighted

as defined in the measurements [28] the number of transactions properly classified as legal, such as the non-detection of weight loss transactions in Table II to IV. The accuracy and sensitivity obtained by RF is 95% and 91%, respectively, as shown in Table V. By comparison, the best accuracy and sensitivity from the tests in this paper is more than 99% for most individual models.

TABLE V
PERFORMANCE COMPARISON WITH RESULTS
EXTRACTED FROM [28]

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

C. REAL-WORLD DATA

The actual credit card data set at a financial institution in Malaysia has been removed from the test. Based on card holders from the South-East Asian region from February to April 2k17. A total of 287,224 transactions were recorded, of which 100+ were classified as fraud cases. Data contains timeline of actions. Compliant with customer privacy requirements, no personal identification information has been discarded. The features distributed in the test are given in Table VI.

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)

A total of 11 features have been removed. The issued codes are found in the standard ISO 8583 [29], and the last two

codes are found in ISO 4217. real numbers, to protect the personal information of customers. The results from the various models are shown in Table VII. All weight accuracy is over 99%, with the exception of SVM at 95.5%. The actual detection weighs NB, DT, and LIR at 100%, and everything else is close to perfection, with the exception of SVM. The best MCCs weigh in from NB, DT, RF, and DS, at 0.990. Fraudulent detection varies from 7.4% LIR to 100% RF, GBT, DS, NN, MLP, and LOR.

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%	80.392%	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

Similar to the bench test, AdaBoost has been released for all individual models. The results are shown in Table VIII. The accuracy and real-time detection weighs the same as those without AdaBoost. AdaBoost helps to improve the detection of weight loss, with the apparent difference in NB, DT, RT, which produces complete weight loss accuracy. The most significant improvement is expressed by the LIR, i.e., from 7.4% to 94.1% accuracy. This clearly demonstrates AdaBoost's usefulness in improving the performance of individual isolates. 1 excellent MCC points issued by NB and RF.

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 multi-voting method is used for the same models extracted from benchmark tests. Results are shown in Table IX. Accuracy and actual detection are weighted in full, or near perfect. DS + GBT, DT + DS, DT + GBT, and RF +

GBT deliver the perfect weight for fraud detection. MCC scores are close to 1 or 1. Most voting results are better than for individual models.

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

Continuing to test the robustness of machine learning algorithms, all real data trials are distorted, by 10%, 20% and 30%. Audio has added to all aspects of the data. Figure 1 shows the detection of fraud on the scale while Figure 2 shows the MCC scores. It is evident that with the addition of noise, the detection of fraud is weighted and the MCC's weight becomes worse as expected. The worst performance, i.e. a significant decrease in accuracy with the MCC, stems from the majority of DT + NB and NB + GBT votes. DS + GBT, DT + DS and DT + GBT show a gradual decrease in performance, but its weighted accuracy is still more than 90% or 30% noise in the data set.

TABLE X
SUMMARY OF THE RELATED WORK

Ref	Authors	Year	Issues	Techniques Used	Remarks
[30]	Altyeb Altaher Taha et.al.	2020	Fraud detection	OLightGBM	Better results
[31]	Surya Narain Kalid et.al.	2020	Anomaly detection	Multiple Classifier System	Works better
[32]	Sara Makki et.al.	2019	Fraud detection	C5.0, SVM, ANN	Devoted ecosystem based on big-data
[33]	Kuldeep Randhawa et.al.	2018	Fraud detection	AdaBoost, Majority voting	Noise is removed
[34]	Andrea Dal Pozzolo et.al.	2018	Fraud detection	Supervised learning	Feedbacks system interaction
[35]	Lorenzo Meneghetti et.al.	2018	Anomaly detection	Isolation forest and local outlier factor	Isolation forest works better on unlabeled data set
[36]	Lutao Zheng et.al.	2018	Fraud detection	Markov-chain model, path based model	Good metrics for accuracy

[37]	Sonali Bakshi	2018	Fraud detection	Hidden Markov model, fuzzy bunching, neural system and data mining	Finds the fraudsters well
[38]	F. Carcillo et.al	2019	Anomaly detection	Hybrid Algorithms using unsupervised learning	Performance is moderate

V. CONCLUSIONS

In this paper, a tutorial on the discovery of credit card fraud with the help of machine learning is provided. Here the best model NB, SVM and DL model is used to look. Publicly available data is used for these types. according to the results of the MCC matrix results used, its results are true and false and a bad prediction. The best MCC rating is 0.823, made up of a large number of votes. At average, a real set of credit card data from a financial institution is used. The same models are unified and integrated into functional. Total MCC points 1 have been using AdaBoost and many voting methods. In addition, 10% to 30% audio is included in the hybrid model test set data. Most voting system is very robust and secure in product performance when a certain sound is added. It shows excellent MCC 0.942 points when 30% noise is added. For further work, the methods learned in this paper will be explained to the online learning models. Some online learning models will be investigated as additional work. The use of online learning will allow for faster detection of fraud cases, which may be real-time. This will help prevent and detect transactions before they happen, as financial losses will be minimized.

VI. FORTHCOMING DIRECTIONS

In the future, we will extend this work to get a few of the interesting schedule with the help of the control and semi-control of the models. We can also make sure that the model of the fixtures of the data to other applications.

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_Report_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 for credit 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 and Bayesian 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 genetic algorithm 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 visualizing the 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: A case 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.

[21] Y. Li, C. Yan, W. Liu, and M. Li, “A principle component analysisbased 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.

[22] 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.

[23] 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.

[24] 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.

[25] 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.

[26] M. W. Powers, “Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation” *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37–63, 2011.

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

[28] 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 eBusiness and Telecommunications*, vol. 4, pp. 335–342, 2017.

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

[30] A. A. Taha, S. J. Malebary, “Intelligent Approach to Credit Card Fraud Detection Using an OLIGHTGBM”, *IEEE Access* (2020), pp. 25579- 25587

[31] S. N. Kalid, K. H NG, G. K Tong, K. C Khore., “A Multiple Classifiers System for Anomaly Detection in Credit Card Data With Unbalanced and Overlapped Classes”, *IEEE Access* (2020), Vol. 8, pp. 28210- 28221

[32] S. Makki, Z. A Assaghir, Y. Taher, R. Haque, M. S Hacid, H. Zeineddine, “An Experimental Study With Imbalanced Classification Approaches for Credit Card Fraud Detection”, *Special Section On Advanced Software And Data Engineering For Secure Societies, IEEE Access* (2019), Vol 7, pp. 93010-93022

[33] K. Randhawa, C. K Loo, M. Seera, C. P Lim, A. K. Nandi, “Credit Card Fraud Detection Using Adaboost And Majority Voting”, *Ieee Access*, (2018) Vol 6, pp 14277-14284

[34] Andrea Dal Pozzolo, Giacomo Boracchi, Olivier Caelen, Cesare Alippi, Gianluca Bontempi, “Credit Card Fraud Detection: A Realistic Modeling and a Novel Learning Strategy”, *Ieee Transactions On Neural Networks And Learning Systems*, (2018) Vol. 29, No. 8, pp. 3784-3794

[35] L. Meneghetti, M. Terzi, S. Del Favero, G. A Susto, C. Cobelli, “Data- Driven Anomaly Recognition for Unsupervised Model-Free Fault Detection in Artificial Pancreas”, *Ieee Transactions On Control Systems Technology*, (2018) pp. 1-15

[36] L. Zheng, G. Liu, C. Yan, C Jiang, “Transaction Fraud Detection Based on Total Order Relation and Behavior Diversity”, *Ieee Transactions On Computational Social Systems* (2018), pp. 1-11

[37] S. Bakshi, “Credit Card Fraud Detection A classification analysis”, *Second International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) IEEE* (2018), ISBN 978-1-5386-1442-6, pp. 152- 156

[38] F. Carcillo, Y.-A. Le Borgne and O. Caelen et al., “Combining unsupervised and supervised learning in credit card fraud detection”,

Information Sciences, Elsevier (2019), pp. 1-15