Credit Card Fraud Detection Based on Machine and Deep Learning

Conference Paper · April 2020

DOI: 10.1109/ICICS49469.2020.239524

Najadat Hassan
Jordan University of Science and Technology
91 PUBLICATIONS **789** CITATIONS
SEE PROFILE

Ola Adnan Altiti
Jordan University of Science and Technology
6 PUBLICATIONS **169** CITATIONS
SEE PROFILE

Ayah Abu Aqouleh
Jordan University of Science and Technology
4 PUBLICATIONS **112** CITATIONS
SEE PROFILE

Mutaz Younes
Jordan University of Science and Technology
7 PUBLICATIONS **122** CITATIONS
SEE PROFILE

Credit Card Fraud Detection Based on Machine and Deep Learning

Hassan Najadat
Computer Information Systems Department
Jordan University of Science and Technology
Irbid, Jordan
najadat@just.edu.jo

Ola Altiti
Computer Science Department
Jordan University of Science and Technology
Irbid, Jordan
olaaltiti@gmail.com

Ayah Abu Aqouleh
Computer Science Department
Jordan University of Science and Technology
Irbid, Jordan
ayaalkhader96@gmail.com

Mutaz Younes
Computer Science Department
Jordan University of Science and Technology
Irbid, Jordan
mohtazscape@gmail.com

***Abstract*—With the rapid evolution of the technology, the world is turning to use credit cards instead of cash in their daily life, which opens the door to many new ways for fraudulent people to use these cards in a bad way. According to the Nilson report, global card losses are expected to exceed $35 billion by 2020. To ensure the safety of users for these credit cards, the credit card's provider should provide a service to protect users from any risk they may face. Consequently, we present our approach to predict legitimate or fraud transactions on the IEEE-CIS Fraud Detection dataset provided by Kaggel. Our model is BiLSTM- MaxPooling-BiGRU-MaxPooling which based on bidirectional Long short-term memory (BiLSTM) and bidirectional Gated recurrent unit (BiGRU). We also applied six machine learning classifiers which are: Naïve base, Voting, Ada boosting, Random Forest, Decision Tree, and Logistic Regression. Comparing the results from machine learning classifiers and our model the results show that our model achieved better as we got 91.37% score.**

***Index Terms*—Fraud Detection, Credit Card Fraud, Deep leaning, Machine learning, BiLSTM, BiGRU, Ada boosting, Voting, Random Forest, Logistic Regression, Naïve base, Max Pooling, Decision Tree**

I. Introduction

Recently, there is a growing usage of the credit card payment method such that most people use credit cards instead of using cash when they do normal payments in their daily life. According to [1], credit cards are found in most Americans wallets. About 7 in 10 Americans have at least one credit card. The credit card allows the customers to track their spending easily and they can know where the money goes. Also, the customers have no limits on their spending, unlike the cash method that is limited to the cash on your wallet. In addition, most companies and institutions now tend to move their business toward online services due to the rapid increase of using modern technology in all fields. Thus, online transactions (e.g. booking a hotel) require a customer to have a credit card to access the services and complete the transaction in such an efficient way that it might be hard and time-consuming to perform while using cash payment. However, a credit card is susceptible to cybercriminals causing credit card fraud. The fraudsters perform fraudulent activities by making unauthorized access to credit card information and such activities cause a financial loss for both company and customer. Thus, the challenges of fraudulent activities increased the demand for systems to detect credit card fraud. The researchers try to build fraud detection systems using machine learning, deep learning, and data mining techniques to detect the transaction whether it is fraudulent transactions or genuine based on datasets that include information about the transactions. However, credit card fraud detection is becoming more complex since the fraudulent transactions for the cards are more and more like legal ones [2]. To solve this issue, credit cards´ providers must use more sophisticated techniques to detect fraud transactions. One of the biggest problems in this field is the lack of good datasets since the datasets available for this problem are imbalanced datasets and have a lot of unknown fields for private insurance. Which makes it harder for the programmers to understand the dataset and build the best model that solves this problem.

In this paper, we present our work to tackle the problem of credit card fraud using machine learning and deep learning models performed on the IEEE-CIS Fraud Detection dataset provided by Kaggle.The remainder of this paper is structured as follows: Section II covers the related work. Section III and IV talks about machine learning and deep learning models. SectionV describes the dataset used in this paper and the preprocessing techniques. Section VI covers the evaluation metrics that used to evaluate the results. Section VII describes the methodology. The last section discusses important findings in our research (Section IX) and concludes the paper with the avenue of future work.

II. Related Work

In this section, we review some previous work related to the Fraud Detection. M. Zareapoor and P. Shamsolmoali in [3] Presented an application based on a bagging ensemble for credit card fraud detection problems. The ensemble approach based on a decision tree algorithm that was used for the experimental step. Moreover, this paper includes a comprehensive study of methods used such as Naïve Bayes (NB), k-Nearest Neighbor (KNN), and Support Vector Machines (SVMs). A real-world credit card dataset was obtained from UCSD-FICO competition used to evaluate their experiments using 10-fold cross-validation techniques. The dataset contains 100,000 records of credit card transactions, including their labels (legitimate and fraudulent). The evaluation measure that used for evaluating the system performance such that Fraud Catching Rate, False Alarm Rate, Balanced Classification Rate, and Matthews Correlation Coefficient. The experimental results show that the bagging classifier based on the decision tree achieved the best performance. R.Patidar and L.Sharma [4] introduced an artificial neural network (ANN) approach with the genetic algorithm to detect fraudulent transactions. The proposed system works when the holder of the credit card uses the card in an unauthorized way; the NN tends to check the pattern that has been used by a fraudster and compare it with the pattern of the original cardholder to ensure if both patterns are a match or not. When there is a big difference between the original pattern and the obtained one, it represents an illegal transaction that will be happened. Several features were used by NN for each transaction that fed the network such as, current transaction descriptor, transaction history descriptor, payment history descriptor, and other descriptors. Moreover, Feed Forward Back Propagation was used as a Learning Algorithm; it is considered as a standard learning technique that employes gradient descent in the error space, which plays an essential role in improving the efficiency. Also, it helps to pick out the parameters for the network such as weight, network type, number of layers, and number of the node. Genetic Algorithm and Neural Network (GANN) as a proposed system aims to detect credit card fraud successfully.

In [5], H.Tran and K.P.Tran used anomaly detection techniques for credit card fraud detection based on the reasons that the detection of fraud must be very flexible in order to track the continuous evolution of fraud over time and the occurrence of unknown anomalies. Also, they proposed two data-driven methods which are one-class support vector machine OCVM with the optimal kernel parameter selection and T2 control chart. The performance of the method tested on large real-time data set of online e-commerce transactions from European credit card holders which contains a total of 284807 non- fraud transactions. Moreover, simulations performed to gen- erate fraudulent transactions, then 284000 transactions were used for training and 200 fraudulent transactions and 200 non-fraudulent for testing. In order to evaluate the results obtained from the methods, they used accuracy, F1-score, Recall (DR), FPR and Precision matrices. The experimental results show that OCVM performs better than T2 control chart with Accuracy= 96.6%, FPR= 8.5% and F-score= 100%. However, the two proposed methods have proven to achieve high accuracy and low false rate in credit card fraud detection.

K.Seeja and M.Zareapoor [6] used a matching algorithm to compare the pattern of a new transaction with existing patterns for each customer, they constructed both fraud and legal patterns for each customer, and when a new transaction comes, they find wheatear it is closer to the fraud pattern or the legal pattern. To find both patterns for each customer they had to separate each customer's transactions, then separate the fraud transactions from legal transactions for each customer, and finally apply the Apriori algorithm to the set of legal and fraud transactions for each customer. Apriori algorithm returns a set of frequent itemsets; they took the largest frequent itemset for both legal and fraud patterns for each customer. Since they have the legal and fraud pattern for each customer, any new transaction from any customer will be compared with both patterns, which make it easier to find if it is fraud or legal transaction.

In [7], A. Roy, J. Sun and R. Mahoney used six classifiers with a dataset before and after the pre-processing phase; The results show a significant improvement when they use the Undersampling technique with the dataset. The dataset used with this paper consist of 284,807 transactions; only 492 of them are fraud transactions. After using the Random Undersampling technique, they changed the ratio to 1:1, which means the number of fraud transactions is the same as the number of legal transactions. They used precision and recall evaluating the classifiers with both datasets; they found out that using the Undersampled dataset increase the precision for all the classifiers significantly.

D.Dighe and S.Patil presented an approach for credit card fraud detection in [8]; it is based on Machine Learning algorithms (ML)- Logistic Regression, K-Nearest Neighbour, Naïve Bayes, Decision Trees and Neural Network algorithms (NN)- Chebyshev Functional Link Artificial Neural Networks (CFLANN), Multi-Layer Perceptron (MLP), applied to highly imbalanced dataset, where the dataset dived into 70% consider as a training set and 30% for testing. Based on the similarity between the history of usage for original cardholders with the current transaction is determining credit card transaction is fraud or genuine. Additionally, to get more accurate results converted dataset from imbalanced dataset into a balanced dataset through a hybrid sampling method. In order to evaluate the results for ML classifiers and NN algorithms, they used performance metrics such as accuracy and sensitivity for ML classifiers and MSE for NN algorithms. The experimental results show that K-Nearest Neighbour was achieved as 99.13% better accuracy than all other ML classifiers, while accuracy in Logistic Regression is 96.27%, Naïve Bayes is 96.98%, and Decision Tree is 96.40. On another side in NN, the MSE for MLP is better than CFLANN.

In [9], M.Puh and L.Brkic presents a comparison of three supervised machine learning algorithms: Random Forest (RF), Support Vector Machine (SVM) and Logistic Regression (LR), they used a dataset that contains credit card transactions made by European cardholders for two days in September 2013. Moreover, the dataset consists of 284,807 samples (transactions), it has 492 fraudulent transactions, 31 features; 28 numerical input variables are a result of Principal Component Analysis (PCA) transformation made by dataset provider, and two non- transformed variables, and finally the class feature. However, the challenges that appear in fraud detection system are highly dealing with an imbalanced dataset, and a non-static environment to overcome these challenges, this paper used SMOTE method, and ensembles and adaptive base learner, respectively. The environment of the fraud detection system is not static, where fraudulent behavior is mutating over time to avoid detection, so the predictive model should not be static. Their experiments were done using two approaches: static and increment, where the evaluated performance based on two measurements namely: AUC and average precision (AP). According to the mentioned results in this paper, SVM has achieved the lowest performance in static and increment setup through AUC and AP, while LR achieved better performance in increment setup through AP measure.

[10], J.O.Awoyemi and A.O.Adetunmbi presents a comparison of three machine learning algorithms namely: naïve bayes, k-nearest neighbor and logistic regression on a dataset that is sourced from ULB Machine Learning Group made in September 2013 by European cardholders, they split it into 70% for training and 30% for testing and validating, the dataset consists of 284,807 transactions and its highly imbalanced and skewed data. Moreover, they used under-sampling and over-sampling methods to avoid dealing with highly imbalanced data, these methods achieve two sets of distribution (10:90 and 34:64), where the ML classifiers are applied to both distributions. In order to evaluate the performance of three classifiers, they used Accuracy, sensitivity, specificity, precision, Matthews correlation coefficient (MCC) and balanced classification rate. The results show from the experiment, the optimal accuracy achieved in 34:66 data distribution using logistic regression is 54.86%, Naïve Bayes is 97.69%, and k-nearest neighbor is 97.92%.

V. Dheepa and R.Dhanapal [11] proposed a model using Support Vector Machine (SVM) with RBF kernel function, where the RBF function is the most flexible function to use with SVM applications, the values of the parameters are se- lected by cross-validation using Grid search. The performance of the SVM is affected by the number of features, which gives a good result when selecting a small number of features for training, so that they selected a small set of features that are relevant to customer behavior (Transaction Amount, Date, Time, Frequency of card usage, Place, Customer ID, and Average amount of transactions per month) for training, these features are transformed into numerical data before used. According to the mentioned results in this paper, accuracy achieved in this model more than 80 percent.

III. Machine Learning

ML considered as a subset of the larger field of artificial intelligence (AI), that teaches a computer to perform a specific task without programming instructions explicitly, where the machine has the ability to learn and improve from experience. ML learning models are widely used in detecting credit card fraud and have proved their efficiency in getting high scores. In our work, we have applied five machine learning classifiers in the dataset which are: Naïve base, Voting, Random Forest, logistic regression, Decision Tree and Ada boosting.

IV. Deep Learning

Deep learning is a subset of machine learning technique that teaches computers to perform tasks which are natural to the human. A computer model learns to perform clas- sification tasks directly from image, text or sound where it builds features automatically based on training data.Deep Learning models including neural networks, convolutional neural networks,Long short-term memory(LSTM) and Gated Recurrent Unit (GRU) provide state of the art results in various classification and prediction tasks.A neural network model used to predict the transactions; the details of the model described in the methodology section.

V. Data Set

The Dataset used for this paper is called IEEE-CIS Fraud Detection. The dataset contains four files, train transaction with 394 columns, train identity with 41 columns, test trans- action with 393 columns, test identity with 41 columns. The identity and transaction files were merged based on transaction id feature; resulting in 433 features in the dataset and 590540 instances. However, there are 378 features contain a lot of null values and thus they were ignored. In addition, the transaction date was deleted since it is not important. The remaining features that contain null values were filled with Nan for categorical features and 0 for numerical features. The dataset is highly imbalanced, such that there are 569875 transactions that are labeled as legitimate transactions, while there are 20663 labeled as fraud transactions which means only 3.626% of the transactions were fraud transactions and this leads the models to predict only legitimate transactions while acting poorly when trying to predict fraud transactions. Thus, it is very important for dataset classes to be balanced when training the models. To solve this problem the following techniques applied to the dataset:

A. Random Under Sampling

In this technique, random instances selected from the majority class and added to the minority class in a way that balances the dataset, which means the majority and minority class ration becomes 1:1. This technique helps us with the dataset.

B. Random Over Sampling

In over sampling technique, the random instances generated by duplicating the instances of the minority class, but using this method will result in overfitting the dataset. In addition, the size of the dataset will increase significantly.

C. Synthetic Minority Oversampling Technique (SMOTE)

SMOTE is an oversampling technique.It relies on the concept of nearest neighbors to create the synthetic data. Rather than duplicating existing samples, it generates data that is similar to the existing ones.

VI. Evaluation metrics

To measure how the proposed model performs, we used different metrics. Since the dataset used in this paper was highly imbalanced, using the accuracy metric alone will not be accurate to measure the performance of the model.

A. Model Evaluation Metrics

Different evaluation metrics were used to evaluate our work in this dataset as shown below.

B. Area Under the Receiver Operating Characteristic Curve (AUC)

AUC score is an implementation dedicated for binary classification or multi-label classification tasks, where it is used the prediction scores to calculate the area under the receiver operating characteristic curve, and then the average would be calculated depending on several ways namely: micro, macro, samples, and weighted, macro set as default. It can be used through sklearn metrics as well we can be calculated as follows:

$$false - positive - rate = \frac{FP}{(FP + TN)}$$

$$true - positive - rate = \frac{TP}{(TP + FN)}$$

1) Precision: Precision is the number of True Positives divided by the number of True Positives and False Positives. We can say it is the number of positive predictions divided by the total number of positive class values predicted.

2) Recall: Recall is the number of True Positives divided by the number of True Positives and the number of False Negatives. We can say it is the number of positive predictions divided by the number of positive class values in the test data. It is also called Sensitivity or the True Positive Rate.

3) F1-Score: F1 score finds the balance between precision and recall.It can be calculated as follows:

$$F1score = 2 * \frac{(precision * recall)}{(precision + recall)}$$

VII. Methodology

We have used different machine learning models including Naïve base, Voting, Random Forest, logistic regression(LR), Decision Tree and Ada boosting on the dataset. To deal with our imbalanced dataset we used three techniques, SMOTE technique, random over sampling and random under sampling. We also used deep learning models on the dataset, we tested different deep learning models and chose the best one which is BiLSTM-MaxPooling-BiGRU-MaxPooling.However, we have built our model that based on BiLSTM and BiGRU models, the architecture of the model described in the Figure1 , as shown there are two inputs to the model which are categorical and numerical features. The model contains embeddings for categorical features and 0.1 of embeddings output dropped to avoid overfitting by using spatial dropout layer. Then the output converted into one dimension via flatten layer and then fed into dropout and spatial dropout layer. As well, the output then passed into BiLSTM and BiGRU layers simultaneously where global max-pooling applied to both models outputs in order to extract the most important features and then the output combined together. On the other hand, numerical features first fed into the drop out layer and concatenated with the output of the categorical feature. For the final prediction, dense layer used with a sigmoid activation function. Also, we have used binary cross entropy loss function and Adam optimizer with a learning rate = 0.01. For performance evaluation, we based on area under ROC curve to evaluate the results.

Table I
THE RESULTS OF USING MACHINE LEARNING MODELS WITH OVER SAMPLED DATASET.

Model	AUC	Precision	Recall	F1_score
Naïve base	71.3%	75.58%	75.58%	54.94
Voting	**80.0%**	83.33%	84.57%	83.95%
Random Forest	79.5%	83.22%	69.76%	75.9%
logistic regression	75.6%	75.72%	73.3%	74.49%
Ada boosting	79.5%	79.82%	72.03%	75.73%
Decision Tree	65.29%	98.03%	100.0%	99.01%

Table II
THE RESULTS OF USING MACHINE LEARNING MODELS WITH UNDER SAMPLED DATASET.

Model	AUC	Precision	Recall	F1_score
Naïve base	71.0%	21.5%	45.97%	26.31%
Voting	**81.8%**	88.9%	70.52%	78.65%
Random Forest	79.8%	84.52%	68.47%	75.65%
logistic regression	80%	76.67%	74.74%	75.69%
Ada boosting	79.5%	80.69%	72.44%	76.34%
Decision Tree	70.06%	78.58%	82.97%	80.72%

VIII. Results

The results from using machine learning models after applying random oversampling technique are shown in table I. As well, table II shows the results from using machine learning models with under sampled dataset. As can be seen, using these techniques did not help with the dataset, the best AUC is 80% and 81% that achieved by hard voting with under and over sampling. Although we have applied SMOTE which is shown in table III. Since the results from using machine learning models were not promising, we considered using deep learning models to achieve better results than what machine learning models achieved. We have used bidirectional Long short-term memory (BiLSTM) with max pooling layer and bidirectional Gated Recurrent Units (BiGRU) with max pooling layer as well; table IV presents the results from the models when using the three sampling techniques.

Table III

THE RESULTS OF USING MACHINE LEARNING MODELS WITH SMOTE.

Model	AUC	Precision	Recall	F1_score
Naïve base	71.0%	10.45%	10.45%	17.07%
Voting	73.1%	57.19%	43.08%	49.14%
Random Forest	74.9%	24.21%	45.09%	31.51%
logistic regression	**80%**	13.01%	72.06%	22.04%
Ada boosting	77.8%	18.89%	55.76%	28.21%
Decision Tree	69.15%	52.28%	61.14%	56.37%

Table IV

THE RESULTS FROM DEEP LEARNING MODELS USING THE THREE SAMPLING TECHNIQUES.

The technique	BiLSTM	BiGRU
Random under sampling	88%	87.4%
Random over sampling	89.07%	90.8%
SMOTE	89.7%	89.2%

As shown the results obtained from machine learning classifiers using random under sampling, SMOTE and random oversampling are not promising, thus, we considered using a deep learning model to achieve better results than what machine learning models achieved. We have used bidirectional Long short-term memory(BiLSTM) with max pooling layer and bidirectional Gated Recurrent Units (BiGRU) with max pooling layer as well; table IV presents the results from the models when using the three sampling techniques. However, we have built our model by concatenating both models. Figure1 describes the model in details. The results from the model are shown in table V.

IX. CONCLUSION

In this paper, we have performed several machine and deep learning models to detect whether an online transaction is legitimate or fraud on the IEEE-CIS Fraud Detection dataset as well built our model which is BiLSTM-MaxPooling-BiGRU-MaxPooling that based on bidirectional LSTM and GRU. We also tested several methods to deal with highly imbalanced datasets including undersampling, oversampling and SMOTE. Set of evaluation metrics used to evaluate the performance of the models. The results from machine learning classifiers show that the best AUC was 80% and 81% that achieved by hard voting with undersampling and oversampling technique. However, the results from machine learning classifiers were not promising compared with our model that achieved 91.37% AUC.

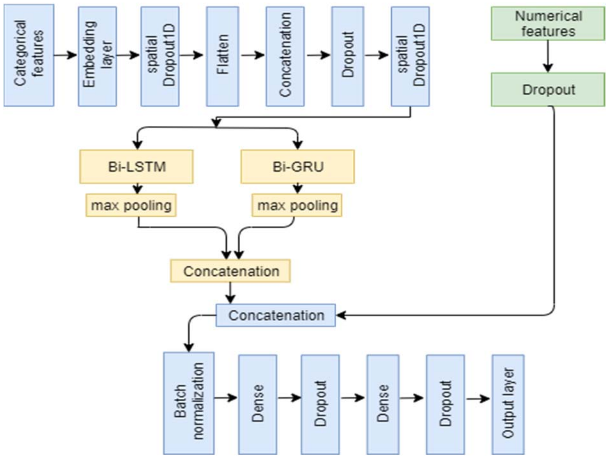

Figure 1. Model architecture

Table V

THE RESULTS FROM OUR MODEL .

The technique	AUC	Precision	Recall	F1_score
Random under sampling	90%	88.07%	80.06%	83.88%
Random over sampling	**91.37%**	91.14%	94.59%	92.81%
SMOTE	90%	58.78%	60.87%	59.81%

REFERENCES

[1] "MS Windows NT kernel description," creditcards.com, accessed: 2010-09-30.

[2] D. Excell, "Bayesian inference–the future of online fraud protection," *Computer Fraud & Security*, vol. 2012, no. 2, pp. 8–11, 2012.

[3] M. Zareapoor, P. Shamsolmoali *et al.*, "Application of credit card fraud detection: Based on bagging ensemble classifier," *Procedia computer science*, vol. 48, no. 2015, pp. 679–685, 2015.

[4] R. Patidar, L. Sharma *et al.*, "Credit card fraud detection using neural network," *International Journal of Soft Computing and Engineering (IJSCE)*, vol. 1, no. 32-38, 2011.

[5] P. H. Tran, K. P. Tran, T. T. Huong, C. Heuchenne, P. HienTran, and T. M. H. Le, "Real time data-driven approaches for credit card fraud detection," in *Proceedings of the 2018 International Conference on E-Business and Applications*. ACM, 2018, pp. 6–9.

[6] K. Seeja and M. Zareapoor, "Fraudminer: A novel credit card fraud detection model based on frequent itemset mining," *The Scientific World Journal*, vol. 2014, 2014.

[7] A. Roy, J. Sun, R. Mahoney, L. Alonzi, S. Adams, and P. Beling, "Deep learning detecting fraud in credit card transactions," in *2018 Systems and Information Engineering Design Symposium (SIEDS)*. IEEE, 2018, pp. 129–134.

[8] D. Dighe, S. Patil, and S. Kokate, "Detection of credit card fraud transactions using machine learning algorithms and neural networks: A comparative study," in *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*. IEEE, 2018, pp. 1–6.

[9] M. Puh and L. Brkić, "Detecting credit card fraud using selected machine learning algorithms," in *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. IEEE, 2019, pp. 1250–1255.

[10] J. O. Awoyemi, A. O. Adetunmbi, and S. A. Oluwadare, "Credit card fraud detection using machine learning techniques: A comparative analysis," in *2017 International Conference on Computing Networking and Informatics (ICCNI)*. IEEE, 2017, pp. 1–9.

[11] V. Dheepa and R. Dhanapal, "Behavior based credit card fraud detection using support vector machines," *ICTACT Journal on Soft computing*, vol. 6956, pp. 391–397, 2012.