Impact Factor (SJIF): 4.542

International Journal of Advance Research in Engineering, Science & Technology

e-ISSN: 2393-9877, p-ISSN: 2394-2444
Volume 4, Issue 8, August-2017

Fraud Detection Technique in Credit Card Transactions using Convolutional Neural Network

Krishna Modi
Student, Computer Department
L D College of Engineering
Ahmedabad, Gujarat, India

Reshma Dayma
Assistant Professor, Computer Department
L D College of Engineering
Ahmedabad, Gujarat, India

***Abstract*— Cashless transactions such as online transactions, credit card transactions, and mobile wallet are becoming more and more popular in financial transactions nowadays. With increased number of such cashless transaction, fraudulent transactions are also increasing. Fraud can be detected by analyzing spending behavior of customers (users) from previous transaction data. If any deviation is noticed in spending behavior from available patterns, there may be chance of fraudulent transaction. To detect fraud behavior, bank and credit card companies are using various methods of data mining such as decision tree, rule based mining, neural network, fuzzy clustering approach, hidden markov model or hybrid approach of these methods. In this paper we have used Convolutional neural network with SMOTE. We have transformed original features into new features.**

***Keywords*— credit card fraud, online fraud, convolution, neural network**

I. Introduction

Fraud can be defined as “Illegal or criminal deception intended to result in financial or personal gain.” Banking fraud can be defined as “The unauthorized use of an individual’s confidential information to make purchases, or to remove funds from the user’s account.” Use of Online Shopping, digital payments, net banking, transactions through payment cards is increasing day by day. Government of India is now also supporting more and more for such type of cashless transactions and e-wallet. As such transactions are increasing, frauds will be definitely going to increased. To prevent such fraudulent transactions, various banks adopt different technology. Root of all this technique is machine learning and data mining. Neural Network is one among them. Data mining plays an important role to detect Financial Fraud learned from historical transaction of customer. Each customer has his/her previous history of transactions. Algorithm learns from customer’s previous history and train a model. When new transaction come, features of new transactions is given to trained model and predicted it as normal or fraudulent one.

II. Literature Survey

Ghosh and Reilly [9] used three-layer feed forward Neural network to detect frauds in 1994. The Neural Network was trained on examples of fraud containing stolen cards, application fraud, counterfeit fraud, Non Received Issue (NRI) fraud, and mail order fraud.

Abhinav and Amlan [6] proposed a Hidden Markov Model to detect the frauds in credit cards. Proposed Model does not require fraud signatures and yet is able to detect frauds by considering a cardholder’s spending habit. This system is also scalable to handle large number of transactions.

Y. Sahin and E. Duman [7] proposed approach to detect credit card fraud by decision tree and Support Vector Machine. Performance of classifier models of various decision tree methods (C5.0, C&RT and CHAID) and a number of different SVM methods (SVM with polynomial, sigmoid, linear and RBF kernel functions) are compared in this study.

An approach is proposed towards credit card fraud detection in [2] using fuzzy clustering and neural network. In this approach fraud detection is done in three phase. First phase is initial user authentication and verification of card details. After successfully completing this phase, fuzzy c-means clustering algorithm is applied to find out normal usage behavior of user based on past transactions. If new transaction is found to be suspicious in this phase, neural network based mechanism is applied to determine whether it was actually fraudulent transaction or an occasional deviation by user.

Kang Fu, Dawei Cheng, Yi Tu, and Liqing Zhang at [3] proposed a convolutional neural network (CNN) based approach to find fraudulent transactions. CNN is a type of feed-forward Neural Network that consist of more than one hidden layer. CNN is a part of deep learning. In this paper, for finding more complex fraud patterns and to improve classification accuracy, a new feature trading entropy is proposed. To relieve the problem of the imbalanced dataset, cost based sampling method is used to generate more number of frauds. Generally, CNN is used for image recognition, Character recognition, image processing, video recognition and recommender system. In this paper for the first time, CNN is used to detect frauds.

III. Problem with credit card fraud detection

One of the biggest problem associated with researchers in fraud detection is lack of real life data because of sensitivity of data and privacy issue. Many researchers have done research with real life data [3], [9], [7], [11] of bank with agreements. To deal with this problem, many tools are available to generate synthetic data.

Second problem is to deal with Imbalance data or skewed distribution because number of fraudulent transactions are very less compare to legitimate transactions. To overcome this problem, synthetic minoring oversampling methods are used to increase number of low incidence data in dataset that generate synthetic fraudulent transactions related with original data set. In [3], cost based sampling is used to generate synthetic fraudulent transactions to balance data set.

Overlapping of data is another problem as some of transactions look like fraudulent transaction, when actually they are legitimate transactions. It is also possible that fraudulent transactions appear to be normal transactions.

IV. Methodology

There are several effective methods to detect banking transaction frauds. Depending on innovative transaction procedures used by frauds, these methods may fail in detecting fraudulent transaction and may cause enormous damage to Card issuers or users. Here, by adding new features in dataset and SMOTE sampling method improves results using Convolutional Neural Network by detecting outlier transactions which can be fraudulent transaction of credit card usage.

Our proposed flow work is divided into two parts.

1. Training phase
2. Prediction phase

In training phase, we will give historical transactions as an input included legitimate and fraudulent. In training phase, Feature Selection of attribute is done and then Synthetic Minority Oversampling Technique (SMOTE) method will be applied to generate synthetic frauds to overcome issue of imbalanced data. We have introduced some features that can be generated from raw features. In order to apply CNN model, we need to transform features into feature matrix to fit the model. Then we will train CNN model.

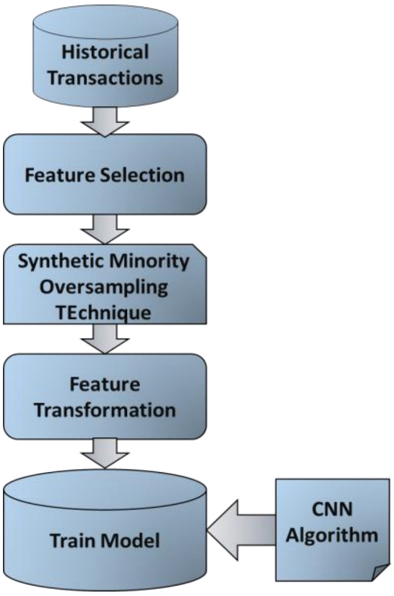


Figure 1Training phase

In Prediction phase, when new transaction will come, it will be given as input. After new transaction's feature extraction and transformation, it will be tested with our CNN trained model classifier. It will be resulted as fraudulent transaction or legislative transaction.

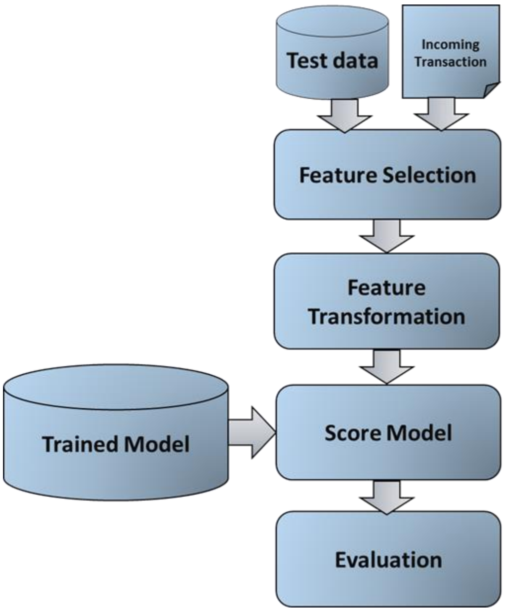


Figure 2 Prediction phase

A. Feature selection

Feature extraction is used to extract useful features to train the model from dataset. These features are extracted from raw data.
User id
Date of transaction
Merchant
Amount
Fraudulent or not

B. SMOTE

One of the problem in fraud detection is to deal with Imbalance data or skewed distribution because number of fraudulent transactions are very less compare to legitimate transactions. To overcome this problem, synthetic minoring oversampling technique is used to increase number of low incidence data in dataset that generate synthetic fraudulent transactions related with original data set.

In [13], SMOTE is used to generate synthetic fraudulent transactions to balance data set. Using this technique, fraudulent samples will be increased, those samples will be given with original transactions to train model.

C. Feature transformation

After selecting features from dataset, pre-processing on data is done and these features are converted as average amount, total amount, number of transactions, bias and trading entropy with reference to current transaction on previous data[3]. For this time window of three day, one week, fifteen days, one month and from beginning of account is taken as shown in figure 3.

Here we have introduced a new feature named bias with merchant.
Avg_Amount_T : Average amount of transactions during past period of time.
NumberT : Total number of the transactions during the past period of time.
TotalAmountT : Total amount of the transactions during the past period of time
BiasAmountT : The bias of the amount of this transaction and AvgAmountT
Trading Entropy: Assume in all transactions of the same customer during the past period of time before the current transaction, there are K kinds of merchant types, the total amount is TotalAmountT, the sum amount of the i-th merchant type is AmountTi(i = 1, 2, . . .,K), the proportion of the i-th merchant type is pi:

$$p_i = \frac{AmountTi}{TotalAmountT} \qquad (1)$$

The entropy of the i-th merchant type can be defined as EntT:

$$\text{EntT}=\sum_i^k pi \log pi \qquad (2)$$

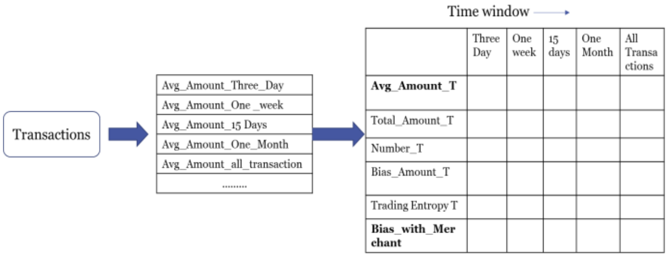
The above calculations only use previous transactions while the current transaction is not involved in. Then we add the current transaction to join the above calculation to obtain the current entropy: NewEntT. So the trading entropy is defined as TradingEntropyT:

$$\text{TradingEntropyT} = \text{EntT-NewEntT} \qquad (3)$$

Merchant Bias: Assume in all transactions of the same customer during the past period of time before the current transaction. Suppose current transaction is done with merchant x with amountX. Merchant X's average amount during previous transactions' is AvgamountX. So bias amount of amountX and AvgamountX is defined as bias_with_merchant. So for i-th merchant in current transaction,

$$\text{bias\_with\_merchent} = \text{amounti} – \text{Avgamounti} \qquad (4)$$

First these features are converted into one dimension, then converted into matrix while training CNN model.
If history of customer is not available, i.e. for a new customer, all values for Avg_amount_T , Total_amount_T, Number_T, Entropy_T would be zero. In this case, model may not able to predict transaction is legitimate or not. To overcome this issue, we averaged all customers who have done transaction in time period T and then have taken bias with that average amount.



Avg_Amount_Three_Day
Avg_Amount_One _week
Avg_Amount_15 Days
Avg_Amount_One_Month
Avg_Amount_all_transaction
....

	Three Day	One week	15 days	One Month	All Transa ctions
Avg_Amount_T					
Total_Amount_T					
Number_T					
Bias_Amount_T					
Trading Entropy T					
Bias_with_Mer chant					

Figure 3 Feature transformation into feature matrices

D. Train Model

For training a model we have used convolutional neural network (CNN). This converted features will be given as input to CNN. Convolutional Neural Network is a part of deep learning. We have used LeNET architecture of CNN. We have used softmax function for neuron and 100 hidden nodes.

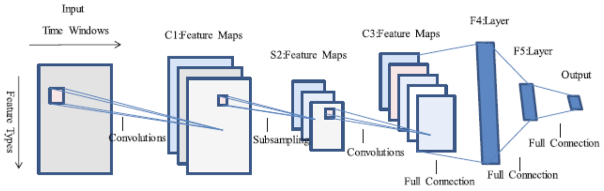


Figure 4 LeNet Architecture

A convolutional neural network consists of several layers. These layers can be of three types.
Convolutional: Convolution means mapping of input layer's neuron to hidden layer (Convolution layer). Each neuron of convolutional layer takes input from previous layer and convolute them to convolutional layer. Taken input should be rectangular grid of neurons. In this, weights specify filter of convolution.
Sub-sampling: Sub sampling layer makes different samples of convolutional layer that is also called as feature map. Subsampling also reduces parameters from previous layer. Average, Maximum are normally used functions for sub sampling.
Fully-Connected: Fully connected layer takes all neurons of previous layer and connect it to every single neuron. After fully connected layer, no convolution is possible.

V. MATRICS TO EVALUATE SYSTEM

As the data is highly imbalance, overall accuracy is not appropriate to evaluate model, since with very high accuracy, almost all fraudulent transactions can be misclassified.
Precision, recall, F1 score, Ratio of True Positive, True Negative, False Positive and False Negative are taken into account for evaluating binary classification.
True Positive (TP) is number of correctly classified fraudulent transactions.
True Negative (TN) is number of correctly classified legitimate transactions.
False Positive (FP) is number of incorrectly classified legitimate transactions.

False Negative (FN) is number of incorrectly classified fraudulent transactions.

Precision (P) = $\frac{TP}{TP+FP}$ (5)

Recall (R) = $\frac{TP}{TP+FN}$ (6)

F1 score is harmonic mean of precision and recall. Value of F1 score lies between 0 to 1. Higher F1 score indicates good model.

F1 score = $2 * \frac{precision*recall}{precision+recall}$ (7)

VI. EXPERIMENT

A. *Description of dataset*

In my research data set used is synthetic data set as real credit card transactions data set is not available due to privacy of customers. I have generated data from [12].
Recourse: https://github.com/metasyn/creditcardfrauddata
Instances: 21300
Attributes:

1. Date
2. Time
3. User id
4. User name
5. Merchant
6. Amount
7. Fraudulent

First this data is split into training set and testing.

Table 1 Split data into training and testing

Dataset	Total	Legitimate	Fraudulent
Total Transactions	21300	20592	708
Training	9300	9041	259
Sample set 1	6000	5783	217
Sample set 2	6000	5768	232

Then SMOTE is applied to training dataset, SMOTE will increase fraudulent samples by making synthetic samples.

Table 2 Summary of training dataset and SMOTE

Details	Instances
Training	9300
Fraudulent Transactions	259
Percentage	2.78%
After SMOTE	
Synthetic frauds	518
Total fraudulent	777
Percentage	8.85%

B. *Environment*

I have used Microsoft azure machine learning studio for implementing CNN classifier. To define CNN, used Net# provided by Azure ML. Azure Machine Learning Studio is a GUI-based integrated development environment for constructing and operationalizing Machine Learning workflow on Azure. I have also used R studio for feature transformation. This transformed features are given as input to CNN classifier to train model.

C. Results

We have experimented using NN and CNN both and then compared results of both.

P= precision
R= recall
NN* = NN with SMOTE
NN**= NN* with added feature
CNN*= CNN with SMOTE
CNN**= CNN* with SMOTE

Table 3 Performance of SMOTE and features with NN and CNN on sample set 1

Method Used	TP	FP	TN	FN	p	r	F1 score
NN	159	2	5781	58	0.988	0.733	0.841
NN*	169	2	5781	48	0.988	0.799	0.871
NN**	171	2	5781	46	0.988	0.788	0.877
CNN	185	27	5756	32	0.873	0.853	0.862
CNN*	187	16	5767	30	0.921	0.862	0.890
CNN**	191	1	5782	26	0.995	0.88	0.934

Table 4 Performance of SMOTE and features with NN and CNN on sample set 2

Method Used	TP	FP	TN	FN	r	p	F1 score
NN	175	5	5763	57	0.972	0.754	0.85
NN*	187	2	5766	45	0.989	0.806	0.888
NN**	192	3	5765	40	0.985	0.828	0.899
CNN	207	34	5734	25	0.859	0.892	0.875
CNN*	207	11	5757	25	0.950	0.892	0.920
CNN**	209	3	5765	23	0.986	0.901	0.941

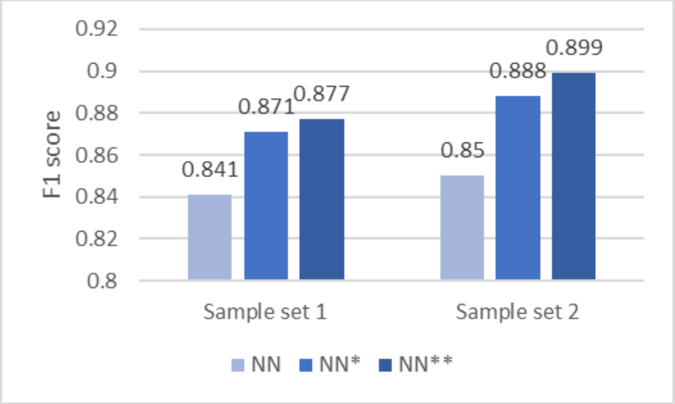


Chart 1 Performance of NN with SMOTE and various features on different Sample sets

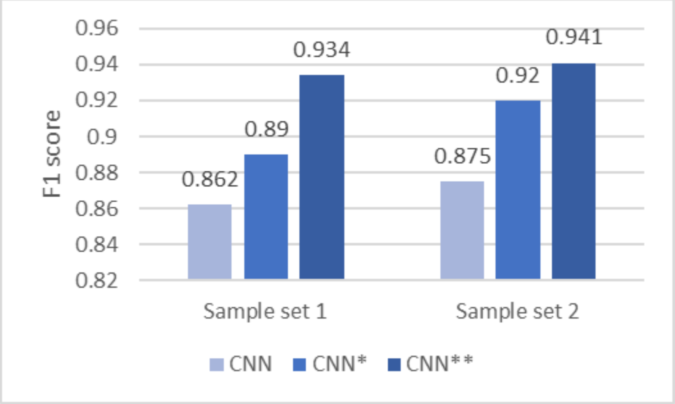


Chart 2 Performance of CNN with SMOTE and various features on different Sample sets

VII. Conclusion

In this paper, Neural network and Convolutional Neural network is applied with SMOTE. We have also transformed features and added new feature that gives better performance by increased number of TP and decreased number of FP. Comparison result shows that in CNN, performance of precision is poor than NN because ratio of legitimate transactions detected as fraudulent one is more than neural network. On other hand in CNN, ratio of detecting fraudulent transaction is more than NN which improves performance of recall and F1 score. Results show that CNN with SMOTE and feature transformation overcome issue of precision and outperforms NN in all terms. Limitation is fraudulent transaction which behaviour is same as legitimate transactions can't be detected.

References

[1] Statista the statistic portal (2017, March 14) available https://www.statista.com/topics/871/online-shopping/

[2] Tanmay Kumar Behera, Suvasini Panigrahi, "Credit Card Fraud Detection: A Hybrid Approach Using Fuzzy Clustering & Neural Network", IEEE Computer Society, 2015

[3] Kang Fu, Dawei Cheng, Yi Tu, and Liqing Zhang, "Credit Card Fraud Detection Using Convolutional Neural Networks", Springer International Publishing AG 2016.

[4] Smt.S.Rajani, Prof.M. Padmavathamma, "A Model for Rule Based Fraud Detection in telecommunications", International Journal of Engineering Research & Technology (IJERT), Vol. 1 Issue 5, July – 2012.

[5] Hidden Markov model (2017, March 15) available https://en.wikipedia.org/wiki/Hidden_Markov_model

[6] Abhinav Srivastava, Amlan Kundu, Shamik Sural, Senior Member, IEEE, and Arun K. Majumdar, Senior Member, IEEE , "Credit Card Fraud Detection Using Hidden Markov Model" , IEEE transactions on dependable and secure computing, vol. 5, no. 1, january-march 2008.

[7] Y. Sahin and E. Duman, "Detecting Credit Card Fraud by Decision Trees and Support Vector Machines", IMECS vol 1, 2011.

[8] Michael Nielsen (2017, March 15), Deep learning available http://neuralnetworksanddeeplearning.com/chap6.html

[9] Ghosh, S., Reilly, D.L.: Credit card fraud detection with a neural-network. In: Proceedings of the Twenty-Seventh Hawaii International Conference on System Sciences, 1994, vol. 3, pp. 621–630. IEEE (1994)

[10] Emin Aleskerov, Bernd fieisleben and Bharat Rao, "CARDWATCH: A Neural Network based database Mining System for Credit Card Fraud Detection"

[11] Sam Maes, Karl Tuyls, Bram Vanschoenwinkel, Bernard Manderick, "Credit card fraud detection using bayesian and neural networks", International Naiso Congress on Neuro Fuzzy Technology, 2002.

[12] Metasyn(2016, October 5), available at https://github.com/metasyn/creditcardfrauddata

[13] Chawla, N.V., Hall, L.O., Bowyer, K.W., Kegelmeyer, W.P. (2002) "SMOTE: Synthetic Minority Oversampling Technique", Journal of Artificial Intelligence Research, vol.16, pp.321-357.

[14] Alejandro Correa Bahnsen, Djamila Aouada, Aleksandar Stojanovic, Björn Ottersten, "Feature engineering strategies for credit card fraud detection", 0957-4174/ 2016 Elsevier.

[15] Smt.S.Rajani, Prof.M. Padmavathamma, "A Model for Rule Based Fraud Detection in telecommunications", International Journal of Engineering Research & Technology (IJERT), Vol. 1 Issue 5, July – 2012.