International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 1 Volume No.6 January – 2022 DOI: 10.46647/ijetms.2022.v06i01.005
ISSN: 2581-4621

Convolutional Neural Network Approach for Mobile Banking Fraudulent Transaction to Detect Financial Frauds

Soumya Shrivastava[1]and Punit Kumar Johari[2]

[1,2]*Madhav Institute of Technology and Science, Gwalior (M. P.), 474009 INDIA*
[1]er.soumyashrivastava@gmail.com
[2]pkjohari@mitsgwalior.in

***Abstract*—In the real world, the identification of financial fraud in compliance with IoT criteria is highly effective since financial fraud causes financial damage. Several forms of financial fraud are likely, but unauthorized usage of mobile payment by credit card no. or certificate no.is the most common scenario. Detection of financial crime is a growing environment in which the victims will keep ahead. However, intelligent fraud detection facets remain scientifically unsupported. Deep learning (DL)arises from the idea of a multi-type representation of the human brain that incorporates basic characteristics at the low level or high-level abstractions.Financial fraud was a big issue as forgers discovered new methods of stealing currency. Therefore, adaptive methods of identification of fraud against forgers are required. Thanks to their versatile nature to detect emergent financial transaction fraud, deep learning approaches were enticing candidates. In this article, we suggest an in-depth learning approach for adapting financial fraud through the use of convolution neural networks (CNN). With the fraudulent transactions dataset, we tested our model experimentally. The results of the analysis show which our methods detect transactional fraud appropriately.**

***Keywords*—Intrusion Detection System, Financial Fraud, Financial Fraud Detection, Information Security, Deep Learning, Restricted Boltzman Machine, Convolution Neural Network.**

I. INTRODUCTION

The identification of fraud is a very difficult task, as it is completely different and has millions of methods. The traditional methods of data analysis were therefore used for a long time for detecting fraud. We are demanding complex and time-consuming tasks that deal with numerous fields of knowledge such as business, finance, economics, & law. In general, in appearance or material, fraud instances can be similar but are not typically identical. The identification of fraud is therefore very difficult [1].

The prevention of financial fraud is vital for the avoidance of frequently catastrophic financial fraud detection (FFD). FFD data from genuine data exposes suspicious acts or behavior and helps decision-makers to formulate effective fraud prevention techniques. It groups compare and summarize the applicable approaches and techniques of financial fraud detection in academic and industrial work published. Secondly, to demonstrate exciting new developments in similar financial adversarial zones, such as the detection of disease and diseases, insider trade, the detection of intrusions, money laundering, and spam detection. [2].

Fig. 1. Fraud Detection Process

Financial fraud in banking, corporate, and government sectors has far-reaching implications. Fraud is a dishonest trick with hopes of monetary benefits. There have been an increased credit card (CC) transactions in high reliance on Internet technology. When online and offline CC transactions developthe main mode of payment, the CCD rate is going up. the rapidly growing problem of financial fraud under IoT is since almost any payment can be made on a mobile device. When mobile companies are expanding rapidly and the IoT world is expanding, financial fraud has been and becomes more prevalent in mobile payments. More than 87% of traders sponsored both desktop and mobile shopping apps [3].

In specific, the purpose of fraud detection is to maximize the right forecasts but maintain an acceptable standard of wrong forecasts. Through minimizing the risk of undetected fraud or false alarms, the high probability of diagnosis can be suggested. There are certain technical terms here. The percentage of true, misdefined transactions is a false alarm (or false positivity). The frequency of fraud detection is the quantity of fraudulently detected transactions.DL is a machine learning substructure based on the structure or operation of the brain known as artificial neural networks [4].

FDS uses anomaly or an outlier detecting technique and uses behavior profiling techniques, whereby the behavioral profile of any person is modeled which observed for any deviation from the norm. Numerous scholars in diverse fields of fraud follow anomaly-based FDS. FDSs based on anomalies can detect new fraud. This is why the literature is

Website: ijetms.in Issue: 1 Volume No.6 January – 2022 DOI: 10.46647/ijetms.2022.v06i01.005
ISSN: 2581-4621

mainly used in the FDS. This approach may be further divided into three types: unregulated, semi-regulated, and controlled identification of anomalies. [5].

Existing detection systems rely on established parameters or learned records that type new patterns of attacks difficult to identify. ML approaches focused on supervised learning or unsupervised learning or deep research of ANN have been extensively researched to discover new trends &realize greater detection accuracy. The most recent approaches used in machinery and in-depth learning financial scams from 2017 to 2020 will be studied in our proposed study. Our work also has a detailed information framework for measuring the performance of FFD transactions.

The remainder of the paper is arranged like this. Similar research present in Section 2 in this area. In Section 3, we present our approach. Section 4 deals with checking. Chapter 5 ends the document.

II. LITERATURE REVIEW

Lichao Sun et al. [2020] Current KOLLECTOR, a new system for detection of images focused on multi-visual bagging DLmethod to collect sequential keyboard tapping knowledge. To authenticate the consumer continually when typing, we are creating a sequence biometric platform. We tested our program empirically utilizing 26 users' real-world contact sessions over eight weeks. We then contrasted our model with the widely employed methods of the shallow machinery and we noticed that our method works better than other strategies and seems to be able to attain an 8.42% error rate, 94.24% accuracy, or 94.41% K-mean by just accelerometer or five keyboard taps. We still use just 3 keyboard taps and consider that the method is already extremely accurate and provides extra chances to make further choices that may contribute to more specific conclusive choices. [6].

S. Mittal and S. Tyagi et al. [2019] In this study, a highly unbalanced dataset used commonly supervised & unsupervised ML algorithms for the order to detect CCF. Unsupervised ML algorithms have been found to manage skewness and to give the best results in classification. CC transactions are now a commonplace, as well as resulting frauds are also prevalent.The most popular forms of fraud detection are illegally collecting card data& utilizes it for online shopping. It is impossible to identify such fraudulent transactions between thousands of regular transactions for CC companies & dealers. When enough data is collected and given to solve this problem, ML algorithms should be utilized [7].

IbtissamBenchaji etal. [2018]Financial fraud crimes have also increased drastically with the growing use of credit card transactions, resulting in a loss of enormous sums in the finance industry. A successful system of fraud prevention has become a must for all banks to mitigate such losses. The identification mechanism for CC fraud is currently a major issue, as -*amount of fraudulent purchases is well below the legal level, the CCfraud data sets are strongly excessive. And there are no minority identity artifacts identified with such distorted data sets for many of the standard classifiers. Firstly, we are proposing a system of sampling focused on the K-means clustering and genetic algorithm for the clustered efficiency of the minority of credit card cases in the unbalanced data collection. In each cluster, we use the genetic algorithm to collect the new samples and to create an efficient fraud detection classifier. We use the K-means algorithm [8].

Yang Kunlin et al. [2018] Propose a modern algorithm known as fraudMemory for fraud detection. This adopts state-of-the-art approaches for portraying consumers or logs of financial networks of other styles more. Our model uses a sequence model innovatively to detention sequence patterns of every transaction & leverages memory networks for performance improvement equally. FraudMemory also has strong adaptability to idea drift by integrating memory components. The empirical research indicates that our model is a possible instrument for the identification of financial fraud [9].

Y. Kunlinetal.[2018] Previous fraud detection experiments deal mainly with specific form transactions and can not be adapted to evolving situations. We suggest a new algorithm named FraudMemory for the detection of fraud here. This adopts state-of-the-art approaches for portraying consumers and logs of financial networks of other styles more. Our model uses a sequence model innovatively to capture sequence patterns of each transaction or leverages memory networks for performance improvement both. FraudMemory also has strong adaptability to idea drift by integrating memory components. The empirical research shows that our model is a possible FFD device. [10].

Aastha Bhardwaj etal.[2018]Propose a text mining system to define and evaluate linguistic data used in financial reporting to detect financial statement fraud. Quantitative & qualitative knowledge is found in the annual reports issued annually by companies. Quantitative data comprises statistics, indicators, averages, and qualitative details comprise of auditors' statements, management reports in a document. NO. The research methods had also been sought via quantitative intelligence to identify a potential cure for financial transaction theft. There has been little to no research on fraud prevention by examining contextual details in financial statements [11].

A. M. Mubalaike and E. Adali [2018]This paper would explain how DL models are helpful for the identification with high precision of fraudulent transactions. Dataset is taken from the legitimate financial data, comprising over six million transactions, of a mobile money service provider in Africa for 1 month. Pre-processed data are used in the most ensemble of decision tree (EDT) collection but in-depth learning strategies as auto-encodes (SAE) and classifications for Restricted Boltzmann machines (RBM). The concert created classifier models areestimated based on accuracy,

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 1 Volume No.6 January – 2022 DOI: 10.46647/ijetms.2022.v06i01.005
ISSN: 2581-4621

responsiveness, specificity. Optimal accuracy figures are 90.49%, respectively, 80.52% or 91.53%. The comparison outcomes show that BRM is superior to other techniques. [12].

N. Balasupramanian etal. [2017]This research is intended to suggest the technologies used to identify and to deter fraudulent electronic purchases utilizing computer learnings and big data analysis. The model enables the storing and cleaning of the vast amount of online transaction info, with the key tool for component analysis being used to remove and the functionality. The decreased features are applied to train ML algo, to recognize the consumer habits in e-transactions, and to classify them.Every e-transaction accepted out by customer-first search algo for correct customer sequence. If a match occurs, the transaction would proceed otherwise a fraudulent transaction is registered. The stored patterns generated by automatic map algo are thus appliedto detect & prevent unauthenticated access to banking transactions [13].

III. PROPOSED METHODOLOGY

An existing technique, the applied RBM deep learning technique is not much efficient in terms of accuracy and performance measures. The Restricted Boltzmann Machines (RBM) functions greedily. A major disadvantage of this greedy algorithm in which a valued lowering procedure is restricted to one lowering run. The fact that the model ignores top-down effects on deference procedure is which interpretation of ambiguous sensory inputs does not adequately account for uncertainties. The key difficulty in the method of fraud detection is discriminating among fraudulent and non-fraudulent practices since false-positive rates minimize the use of electronic payment processes. We use CNNs with expectations to identify fraud in compliance with changing environmental conditions.

We would apply DL to the grouping of anomaly-based NIDS with a real-time financial database to avoid such recurrent problems and equate the identification findings with the previous study.

A. Restricted Boltzmann Machines (RBM)

RBMs are an algorithm that helps to classify, reduce dimensionality, map, reverse, collaborative filter, and learn about features. RBM isa probabilistic graphic model signified by deep neural stochastic networks. RBMs were made more useful in many machine learning problems by increasing computer capacity and developing faster learning algorithms. Boltzmann Machinery (BM), in which the power function in their free parameters is linear, is the linear form of the Marko random field (MRF). The distribution is sufficiently complicated by adding hidden nodes. By incorporating more hidden variables, we may increase the simulation performance of the Boltzmann engine. Boltzmann's restricted machines are BMs deprived of obvious, secret links; hence, term' limited'. The following shows a graphic description of an RBM [14].

Energy function E (v, h) is specified by:

$$E(v,h)= – b'v – c'h – h'Wv \qquad (1)$$

Here, E is energy from RBM. W stands for weights amongst hidden & visible units or b; c signifies the balance of visible & hidden strata.

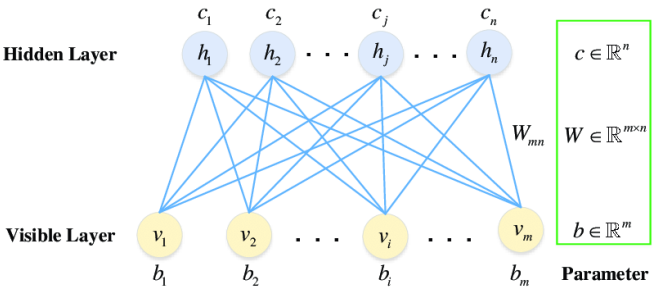

Fig.2.RBM Graphical View

With respect to free energy, eq. (1) shall be provided by:

$$F(v) = – b'v – \sum_i \log \sum_{hi} e^{h_i (c_i + W_i v)} \qquad (2)$$

RBMs are autonomous and can be written in the following line:

$$p(h \mid v) = \prod_i p(h_i \mid v) \qquad (3)$$

$$p(v \mid h) = \prod_j p(v_i \mid h) \qquad (4)$$

The free energy now comes from:

$$F(v) = – b'v – \sum_i \log(1+e^{(ci + Wi v)}) \qquad (5)$$

The log-like gradients of RBM are as follows when we combine (1) and (5):

$$E_v[p(v_i \mid h)] - v_j^{(i)} (6)$$

B. Convolution Neural Networks

CNN is a widely-used paradigm for deep learning inspired by animals' visual cortex. Convents are like ordinary neural networks and may be viewed as an acyclic neuron array. A neural network varies primarily from it a neuron of the secret layer is linked only in the previous layer with a subset of neurons. Owing to this sparse connectivity, it can learn implicit characteristics. The deep network design outcomes in a hierarchical extraction function, i.e. qualified first layer filters are recognizable as a series of edges or color blobs, 2nd layer as figures, next layer filters can recognize object pieces, & final layer filters may classify the objects. objects [15].

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 1 Volume No.6 January – 2022 DOI: 10.46647/ijetms.2022.v06i01.005
ISSN: 2581-4621

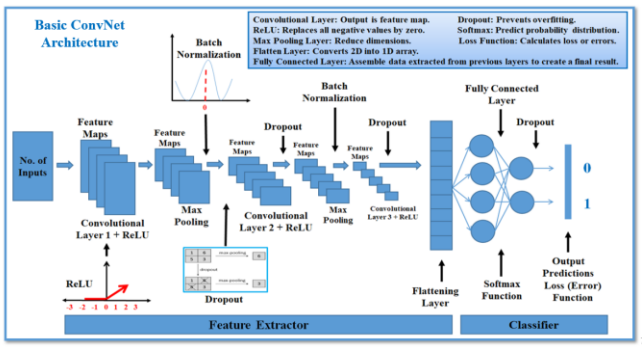

Fig.3. Basic ConvNet architecture

1) Convolutional Layer

This layer constitutes a fundamental unit of a ConvNet, which includes calculations. It is a set of signature maps of neurons. The layer parameters are a series of filters or kernels that can be analyzed. Such filters are complemented by the feature maps to generate a two-dimensional activation map that shows the output volume as it is stacked along the depth axis. Weight (parameter-sharing) of neurons in the same map decreases network size by the number of parameters. A hyperparameter called the receptive field is a spatial extension of sparse relation amid two-layer neurons. The hyperparameters that monitor output volume size include depth(number of layer filters), stride(filter movement) & zeropaddings (to monitor output space). The ConvNets are equipped with rear propagation and reverse is also provided by convolution but spatially reverse filters. [15].

2) Rectified Linear Unit(ReLu)

ReLu is a non-lining function that substitutes for zero in the feature map for all negative pixel values. ReLU's purpose is not to be linear in the CNN, because the rest of this knowledge from the real world we have to know will not be linear. [15].

3) Pooling Layer

Basic ConvNet architecture has alternate conv or group layers so these functions minimize (without loss of information) spatial dimensions of activation maps & no. of parameters in a network, thus minimizing total device complications [16]. This tests the overfitting issue. The bundling operations include max pooling, average pooling, stochastic pooling, spectral reselling, spatial pyramid bundling, or multi-scale orderless pooling.

Maxpooling: It is often referred to as subsampling or downsampling or space pooling. It reduces the dimensionality of each characteristic map but contains the most important information. Various forms may be max-pooling such as number, mean, and average. In the case of Max Pooling, the most significant aspect from the rectified function map of that window is a spatial neighborhood (for example, a 2 x 2 window). We may also take the standard (average pooling) or all the components in the window instead of making the largest component. Max pooling appears to work best from time to time.

4) Fully Connected Layer(FCL)

For classification operations, FCL is applied. For classification, it uses the softmax activation feature. The word "Fully Connected" implies that in the layer above every neuron is related to each neuron in the layer below. The findings from the previous layers are better. This layer is primarily concerned with classifying the input picture based on the higher-level characteristics [16].

$$f(x)=\max(0, x)$$

5) Loss Layer

FCL is a loss layer that measures loss or error or penalizes deviations between expected and the actual output. Softmax loss is used to estimate a single class from K that is equally exclusive. It is a normal feature of failure. The logistic equation is multinomial. It maps projections to non-negative values & is generalized to achieve distribution of probability over groups. Calculating Hinge failure is a wide margin classifier, Vector Machine Support. Euclidean failure should be used to revert to real-value labels [16].

C. Proposed Algorithm

Step 1. Start
Step 2. Collect the Mobile Banking fraudulent transaction dataset by Paysim for a month.
Step 3. Perform data preprocessing to extract relevant features.
Step 4. Normalize the values of the features on the scale of 0 to 1 by

$$Z=(x-\min(x))/(\max(x)-\min(x))$$

Step 5. Hot encoding is applied on categorical features to get splitted form
Step 6. Apply CNN on the preprocessed features to achieve higher accuracy
1) ConvLayer capture the Low-Level features
2) ReLU LayerConverts all negative values to 0 and keeps positive values similar
3) Pooling layer reducesthe spatial size of Convolved Feature
4) Flattening converts the data into a 1-dimensional array for inputting it to the next layer
5) Fully-connected layer learns non-linear high-level function combinations, as defined by convolution layer performance
Step 7. Check whether a transaction is fraudulent
Step 8. Measure performance
Step 9. End

D. Proposed Flowchart

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 1 Volume No.6 January – 2022 DOI: 10.46647/ijetms.2022.v06i01.005
ISSN: 2581-4621

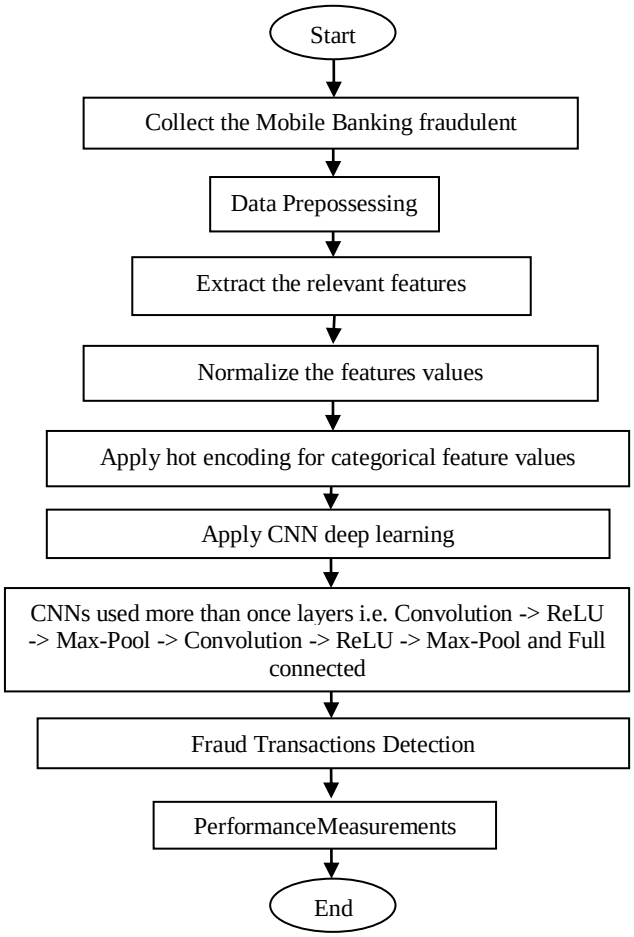

Fig. 4. Proposed Model

IV. EXPERIMENTAL ANALYSIS

Experimental studies were performed to analyze the feature selection accuracy on a given dataset. The simulator was programmed using Python. Original features with their description forgiven dataset are shown in Table I.

A. Dataset Description

There is a small range of publicly available finance data sets, mainly in the developing field of mobile payment identification. Paysim[17], a virtual monetary financial simulator, has gathered private data to create a simulated dataset. We used fraudulent transactions from mobile banking, consisting of 6 billion transactions over a month.

Table I: Original features with their description

Features	Description
Step	1 step is 1 hour. (30 days simulation)
Type	Cash_in, Cash_out, Debit, Payment, Transfer
Amount	Amount of transactions in local currency
NameOrig	Customer who started a transaction
OldBalanceOrg	Early balance previous transaction
NewBalanceOrg	New balance afterward transaction
NameDest	Customer who is a recipient of a transaction
OldBalanceDest	Early balance recipient before a transaction
NewBalanceDest	New balance recipient after a transaction
Class	Fraud, Genuine (1,0)
IsFlaggedFraud	A transaction whose amount more than 200000 (1,0)

Table II indicates extracted new features after preprocessing. In this, 2 new features ErrorBalanceOrg and ErrorBalanceDest are generated for each transaction of originating and destination accounts to differentiate between fraudulent and genuine transactions.

Table II: Created Feature after data preprocessing

Features	Description
Step	1 step is 1 hour. (30 days simulation)
Type	Cash_out, Transfer (1,0)
Amount	Amount of transactions in local currency
OldBalanceOrg	Initial balance before a transaction
NewBalanceOrg	New balance afterward transaction
OldBalanceDest	Initial balance recipient before a transaction
NewBalanceDest	New balance recipient afterward transaction
Class	Fraud, Genuine (1,0)
ErrorBalanceOrg	NewBalanceOrg+ Amount-NewBalanceOrg
ErrorBalanceDest	OldBalanceDest+ Amount-NewBalanceDest

Table III: splitted 4 different features after using one-hot encoding

Type		class	
Cash Out	Transfer	Fraud	Genuine
0	1	0	1
1	0	1	0

B. Screenshots of simulated result

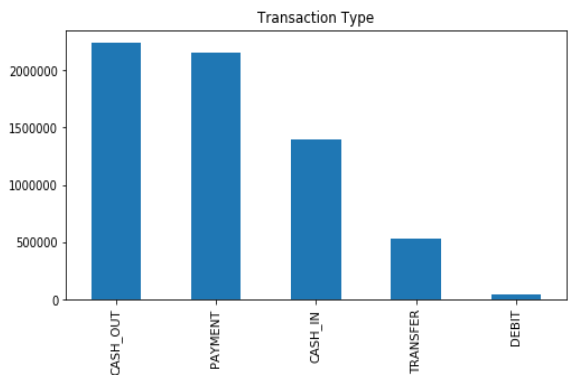

Fig. 5.Before preprocessing type of transaction

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 1 Volume No.6 January – 2022 DOI: 10.46647/ijetms.2022.v06i01.005
ISSN: 2581-4621

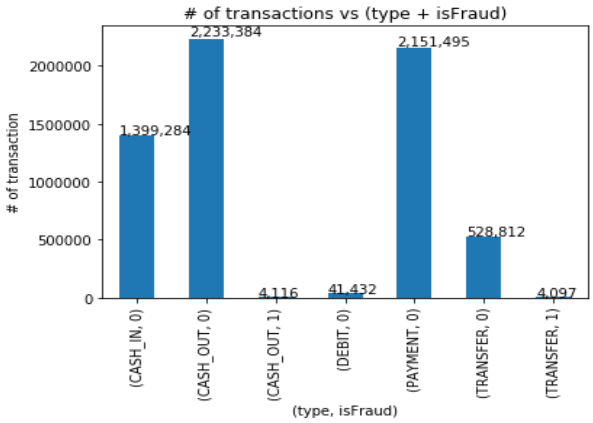

Fig. 6. Before preprocessing to check type is fraud

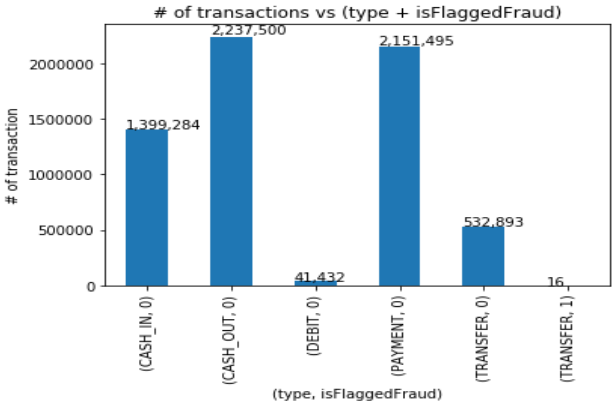

Fig. 7. Before preprocessing to check type isFlaggedFraud

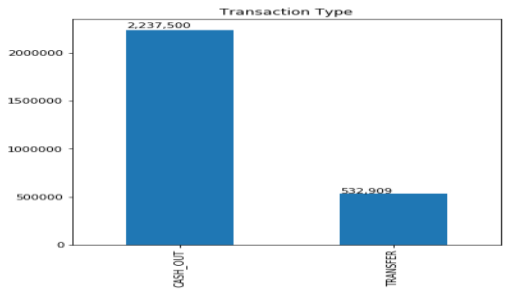

Fig. 8. After preprocessing to check transaction type

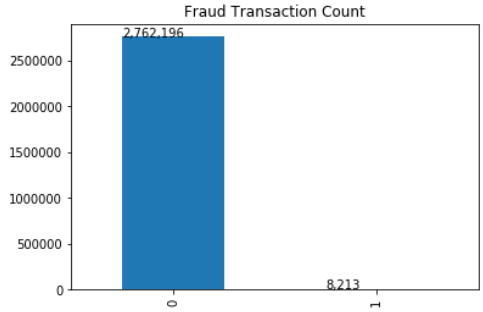

Fig. 9. After preprocessing to check fraud transaction count

	step	type	amount	oldbalanceOrg	newbalanceOrig	oldbalanceDest	\
0	1	TRANSFER	181.00	181.0	0.0	0.0	
1	1	CASH_OUT	181.00	181.0	0.0	21182.0	
2	1	CASH_OUT	229133.94	15325.0	0.0	5083.0	
3	1	TRANSFER	215310.30	705.0	0.0	22425.0	
4	1	TRANSFER	311685.89	10835.0	0.0	6267.0	

	newbalanceDest	isFraud	ErrorBalanceOrig	ErrorBalanceDest
0	0.00	1	362.00	181.0
1	0.00	1	362.00	21363.0
2	51513.44	0	244458.94	182703.5
3	0.00	0	216015.30	237735.3
4	2719172.89	0	322520.89	-2401220.0

```
cat_columns = ["type", "isFraud"]
df_processed = pd.get_dummies(used_data, prefix_sep="_", columns=cat_columns)
df_processed.rename(columns = {'type_CASH_OUT':'cash_out','type_TRANSFER':'tran
print(df_processed.columns)
print(df_processed)
```

Fig. 10. After generating New features

	step	amount	oldbalanceOrg	newbalanceOrig	oldbalanceDest
0	0.000000	0.000002	0.000003	0.000000	0.000000
1	0.000000	0.000002	0.000003	0.000000	0.000059
2	0.000000	0.002479	0.000257	0.000000	0.000014
3	0.000000	0.002329	0.000012	0.000000	0.000063
4	0.000000	0.003372	0.000182	0.000000	0.000018
5	0.000000	0.001194	0.000451	0.000000	0.000811
6	0.000000	0.000616	0.000033	0.000000	0.000197
7	0.000000	0.000058	0.000000	0.000000	0.001833
8	0.000000	0.000252	0.000343	0.000000	0.000072
9	0.000000	0.000677	0.001328	0.000333	0.000001
10	0.000000	0.000897	0.000051	0.000000	0.000372
11	0.000000	0.000513	0.003517	0.003269	0.000146
12	0.000000	0.001481	0.002720	0.000508	0.000612
13	0.000000	0.001020	0.000423	0.000000	0.000280
14	0.000000	0.000462	0.000174	0.000000	0.000163
15	0.000000	0.000843	0.000000	0.000000	0.000267
16	0.000000	0.000186	0.000000	0.000000	0.000069
17	0.000000	0.000852	0.000000	0.000000	0.000291

Fig. 11. After Feature Normalization

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv1d (Conv1D)              (None, 9, 32)             96
_________________________________________________________________
batch_normalization (BatchNo (None, 9, 32)             128
_________________________________________________________________
dropout (Dropout)            (None, 9, 32)             0
_________________________________________________________________
conv1d_1 (Conv1D)            (None, 8, 64)             4160
_________________________________________________________________
batch_normalization_1 (Batch (None, 8, 64)             256
_________________________________________________________________
dropout_1 (Dropout)          (None, 8, 64)             0
_________________________________________________________________
flatten (Flatten)            (None, 512)               0
_________________________________________________________________
dense (Dense)                (None, 64)                32832
_________________________________________________________________
dropout_2 (Dropout)          (None, 64)                0
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 65
=================================================================
Total params: 37,537
Trainable params: 37,345
Non-trainable params: 192
_________________________________________________________________
```

Fig. 12. Proposed CNN Parameters

```
Train Confusion Matrix
[[2071621      43]
 [   2141    4001]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00   2071664
           1       0.99      0.65      0.79      6142

    accuracy                           1.00   2077806
   macro avg       0.99      0.83      0.89   2077806
weighted avg       1.00      1.00      1.00   2077806

Train Accuracy : 99.89488912824392 %
Train Sensitivity : 98.93669634025717 %
Train Specificity : 99.89675768000377 %
```

Fig. 13. Confusion matrix for Training data

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 1 Volume No.6 January – 2022 DOI: 10.46647/ijetms.2022.v06i01.005
ISSN: 2581-4621

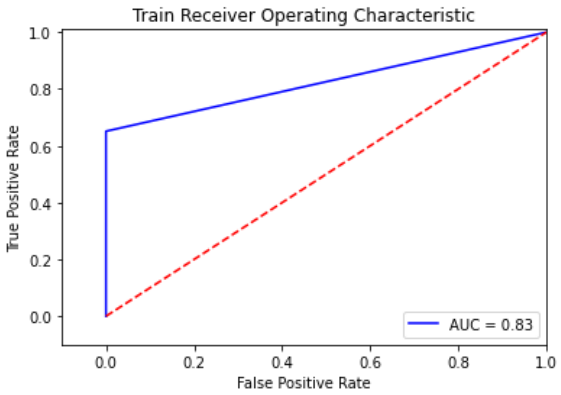

Fig. 14. ROC for Training data

```
Test Confusion Matrix
[[690515     17]
 [   702   1369]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    690532
           1       0.99      0.66      0.79      2071

    accuracy                           1.00    692603
   macro avg       0.99      0.83      0.90    692603
weighted avg       1.00      1.00      1.00    692603

Test Accuracy : 99.89618872572021 %
Test Sensitivity : 98.77344877344876 %
Test Specificity : 99.89843999785884 %
```

Fig. 15. Confusion matrix for Testing data

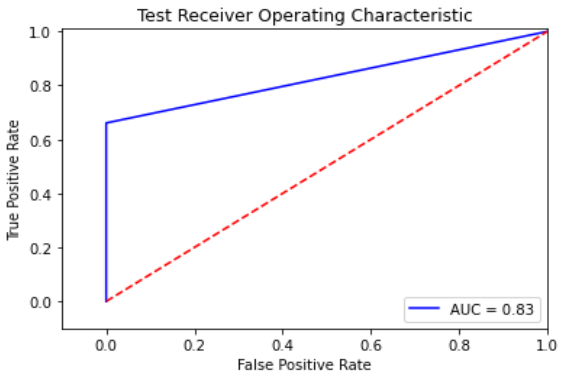

Fig. 16. ROC for Testing data

C. Comparison of Graphical representation of data

Table IV: Comparison of detection rate accuracy results

Dataset	Accuracy (%)	
	RBM	CNN
Training set	96.18	99.89
Test set	96.19	99.89

The proposed CNN based fraud detection system achieved around 99.89% accuracy for only 25% testing data out of total dataset and 100 iterations.

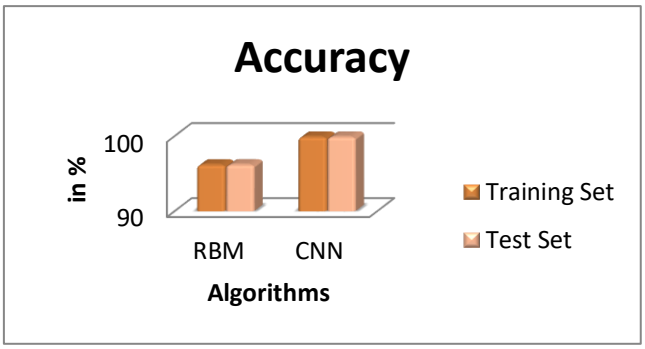

Fig. 17. Accuracy (Detection rate) comparisons on Deep learning methods for train & test data

The result analysis is the comparison between existing RBM and proposed CNN based on accuracy. The proposed technique gives better, accurate detection results. In fig. 17, shows the Accuracy Detection comparisons on Deep learning methods for train & test data.

Table V: Comparison of Sensitivity results

Dataset	Sensitivity (%)	
	RBM	CNN
Training set	96.20	99.94
Test set	96.20	99.77

The result analysis is the comparison between existing RBM and proposed CNN on basis of Sensitivity. The proposed technique gives the highest Sensitivity results. In fig. 18, shows the Sensitivity comparisons on Deep learning methods for train & test data.

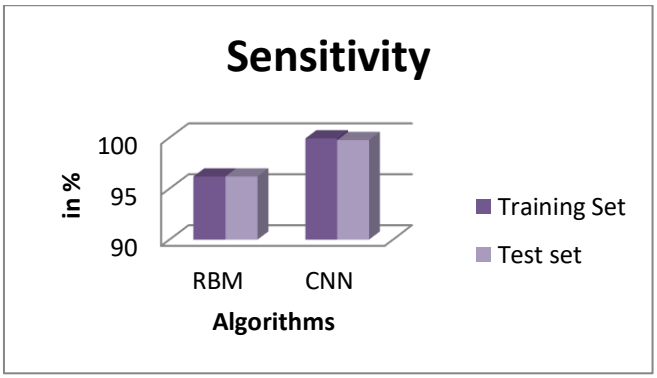

Fig. 18. Sensitivity comparisons on Deep learning methods for train & test data

Table VI: Comparison of Specificity results

Dataset	Specificity (%)	
	RBM	CNN
Training set	81.6	99.89
Test set	84.2	99.89

The result analysis is the comparison between existing RBM and proposed CNN on basis of Specificity. The proposed

technique gives lower Specificity results. In fig. 19, shows the Specificity comparisons on Deep learning methods for train & test data.

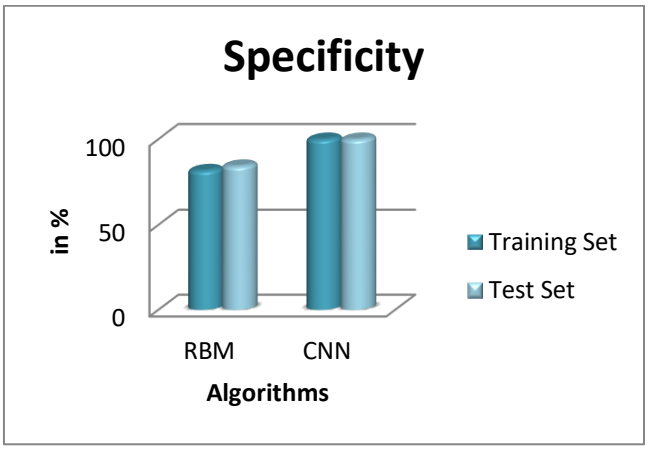

Fig. 19. Specificity comparisons on Deep learning methods for train & test data

V. CONCLUSION

Since the beginning of financial institutions, fraud has always been a concern. While the techniques of fraud have evolved, the primary goal for these institutions remains to secure financial transactions and also to detect fraud. The number of credit cards or their prevalent usage on-line transfers has rendered credit card theft, amongst other illegal practices in the financial sector, a major problem. One of the biggest problems in the prevention of fraud is the convergence of dishonest and honest transactions. In this paper we use a method of neural networking, the main contribution of this paper, to detect fraudulent transactions. Our method of detecting adaptive fraud has been tested quantitatively. Our findings indicate that fundamental forms of learning appear to be strong candidates for financial fraud. The proposed CNN based fraud detection system achieved around 99.89% accuracy for training data and 99.89% accuracy for test data.

References

[1] Muhammad Arif and Amil Roohani Dar, “Survey on Fraud Detection Techniques Using Data Mining”, International Journal of u- and e-Service, Science and Technology Vol.8, No.3 (2015), pp.163-170.

[2] Pankaj Richhariya, “A Survey on Financial Fraud Detection Methodologies”, International Journal of Computer Applications (0975 – 8887) Volume 45– No.22, May 2012.

[3] https://pdfs.semanticscholar.org/debd/146f274fb4a4f92885640698b3220995d75c.pdf

[4] Dahee Choi and Kyungho Lee, “An Artificial Intelligence Approach to Financial Fraud Detection under IoT Environment: A Survey and Implementation”, Hindawi Security and Communication Networks Volume 2018, pp. 1-15. https://doi.org/10.1155/2018/5483472.

[5] Abdallah, A., Maarof, M. A., & Zainal, A. (2016). Fraud detection system: A survey. Journal of Network and Computer Applications, 68, 90–113. doi:10.1016/j.jnca.2016.04.007.

[6] L. Sun *et al*., "KOLLECTOR: Detecting Fraudulent Activities on Mobile Devices Using Deep Learning," in *IEEE Transactions on Mobile Computing*. doi: 10.1109/TMC.2020.2964226.

[7] S. Mittal and S. Tyagi, "Performance Evaluation of Machine Learning Algorithms for Credit Card Fraud Detection," *9th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, Noida, India, 2019, pp. 320-324.

[8] D. Varmedja, M. Karanovic, S. Sladojevic, M. Arsenovic, and A. Anderla, "Credit Card Fraud Detection - Machine Learning methods," *18th International Symposium INFOTEH-JAHORINA (INFOTEH)*, East Sarajevo, Bosnia and Herzegovina, 2019, pp. 1-5.

[9] A.Mishra and C. Ghorpade, "Credit Card Fraud Detection on the Skewed Data Using Various Classification and Ensemble Techniques," *IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS)*, Bhopal, 2018, pp. 1-5.

[10] Kunlin, Y. (2018). A Memory-Enhanced Framework for Financial Fraud Detection. 17th IEEE International Conference on Machine Learning and Applications (ICMLA). doi:10.1109/icmla.2018.00140.

[11] A. Bhardwaj and R. Gupta, "Qualitative analysis of financial statements for fraud detection," *International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, Greater Noida (UP), India, 2018, pp. 318-320. doi: 10.1109/ICACCCN.2018.8748478.

[12] M. Mubalaike and E. Adali, "Deep Learning Approach for Intelligent Financial Fraud Detection System," *3rd International Conference on Computer Science and Engineering (UBMK)*, Sarajevo, 2018, pp. 598-603. doi: 10.1109/UBMK.2018.8566574.

[13] N. Balasupramanian, B. G. Ephrem, and I. S. Al-Barwani, "User pattern-based online fraud detection and prevention using big data analytics and self-organizing maps," *International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, Kannur, 2017, pp. 691-694. doi: 10.1109/ICICICT1.2017.8342647.

[14] Fischer, A., &Igel, C. (2012). An Introduction to Restricted Boltzmann Machines. Lecture Notes in Computer Science, 14–36. doi:10.1007/978-3-642-33275-3_2.

[15] Rahul Haridas and Jyothi R L, “Convolutional Neural Networks: A Comprehensive Survey”, International Journal of Applied Engineering Research, Volume 14, Number 3, 2019, pp. 780-789.

[16] N. Aloysius and M. Geetha, "A review on deep convolutional neural networks," *International Conference on Communication and Signal Processing (ICCSP)*, Chennai, 2017, pp. 0588-0592. doi: 10.1109/ICCSP.2017.8286426.

[17] https://www.kaggle.com/ntnu-testimon/paysim1/data