A Novel Trio-Hybrid for Detecting Fraudulent Credit Card Transactions

Sarika Jain[1], Shripriya Dubey[1], Namrata Tiwari[1], Yashvi Jain[1] and Atef Shalan[2]

[1]*National Institute of Technology Kurukshetra, Haryana, India*
[2]*Georgia Southern University, Georgia, United States*

Abstract

In this era of digitization, credit card frauds shake down the spirits of not only the customers but also the merchants, which incurs a loss of billions of dollars globally. To combat such frauds, a robust and responsive system is needed that can flag the fraudulent transaction instantly before it happens. The existing systems are great at detecting and battling with fraud after it has happened but slouch in case of prevention of such crimes. They aren't good¬¬ at optimization and also struggle in terms of response time. The inefficiency of existing systems is attributed to either working on a single machine learning technique, or just combining two of them. We present a Trio-Hybrid of K-means, Genetic algorithm, and artificial neural network approaches to deal with the aforementioned problems. The K-means algorithm helps in reducing the training time of neural networks and the genetic algorithm helps in feature selection to prevent the neural network from being over-trained, thereby reducing the cost of the system. We leverage the benefits provided by these three techniques and put them together into a trio for the first time and achieve an accuracy of 99.94% in detecting the fraudulent credit card transactions.

1. Introduction

The unauthorized and ill-intended use of credit cards to commit a crime and causing monetary harm to its owner is defined as Credit Card fraud [1]. Frauds with credit cards contribute to a major part in the domain of crime through digital resources. As digitalization is spreading its roots in the global arena, the bulk of transactions take place via credit cards all over the world. For this market to thrive, the credibility of credit cards is mandatory. As with the hike in the number of credit card users, frauds with Credit Cards also rise over the globe. They cause the loss of billions of dollars to companies and customers and impose a huge loss on the growth of any business; and if not controlled they might dump harm on the country's economy [2]. The mannerism of committing credit card fraud keeps evolving and changing due to tech-savvy and shrewd fraudsters; hence, there must be devised a way to put a leash on the ever-evolving fraud techniques to save the world from huge economic losses. There are many ways in which credit card fraud can be carried out such as skimming, stealing, robbing the details, putting chips in

ACI'22: Workshop on Advances in Computation Intelligence, its Concepts Applications at ISIC 2022, May 17-19, Savannah, United States

*Corresponding author.
jasarika@nitkkr.ac.in (S. Jain)
https://sites.google.com/view/nitkkrsarikajain/ (S. Jain)
0000-0002-7432-8506 (S. Jain)

the ATM, cloning, phishing, spying on data from the merchant's system, and erasing the old data on-chip being a few of them. The pattern of credit card fraud is dynamic as tech-savvy fraudsters pose typical challenges in curbing their acts [3]. A good and effective system for the detection of fraudulent credit card transactions must have the robustness and adjustability factor to the changing environment otherwise it might prove to be useless and futile [4]. The characteristics of a good and effective fraud detection system are (a) Accuracy, i.e. the frauds detected by the system must be correct, (b) The fraudulent transaction must be detected while it's being processed (or while it's in transit) and not after its completion, and (c) The system must not misconceive the non-fraudulent or genuine transactions as fraud [5]. Here we list the envisioned applications and use cases of such a fraudulent credit card transaction detection system and also the challenges involved in modelling such a system.

Applications and Use Cases of a Credit Card Fraud Detection System

1. Any terminal that provides the electronic hardware used to swipe cards such as the Point of Service terminals at retail stores.

2. Assistance at the merchants' site for example merchants like insurance companies will be able to identify a fraud client easily.

3. Check at the bank's end. Banks may take a step on detecting fraud taking place associated with an account in their bank.

4. Real-time software systems for example e-ticketing services.

5. Websites that provide e-commerce services such as online shopping etc.

Challenges Involved in modelling a Fraud Detection System

1. Inaccessible data or lack of availability of data. The data associated with a customer's credit card and account information is highly confidential hence no bank or company easily avails the dataset of their customers. As a result, the data is not readily available.

2. Imbalance in the data i.e. the number of fraud transactions is very less as compared to the genuine ones.

3. The behavior of a fraudulent profile keeps changing and its nature is dynamic.

4. The time taken by the system to decide whether a transaction is fraud or not must be very less.

5. Overlapping of transactions i.e. a genuine transaction's nature is very similar to a fraud one.

6. Features and parameters to be processed are very large in number.

7. Selecting the optimal parameters or features is a demanding and challenging task.

8. The noise in the data needs to be managed and altered.

The pattern followed by all the fraudulent transactions is generally very similar and we can categorize some transaction as fraud using any of the following pattern recognition systems as K-Nearest Neighbour (KNN), Artificial Neural Networks, Fuzzy Logic Based System, Artificial Immune System, Naïve Bayesian Network, Hidden Markov Model, Support Vector Machine (SVM), Decision Trees, Ensemble Classifier and Logistic Regression. We should know their advantages and disadvantages in order to leverage the benefits, when some technique is chosen to be applied. Table 1 shows the certain advantages and disadvantages of various techniques for credit card fraud detection.

This work presents a novel trio-hybrid of artificial neural network, k-means clustering, and genetic algorithm (GA) that can precisely and accurately detect and prevent fraudulent credit card transactions while they are in transit. The major contributions and the objectives of the proposed work are (i) to minimize the time required to train the neural network system by using k-means clustering, (ii) utilizing a genetic algorithm to prevent the system from being over-trained, thereby reducing the cost of the system. We have provided an algorithm for the trio-hybrid of the three mentioned approaches and have found an Accuracy of 99.94% and Loss Value of 0.561%. The paper has the following sections: Section II is the Literature Review of the noteworthy existing works that are parallel to the proposed solution and a comparative analysis between them followed by some of the benchmark systems in credit card fraud detection. Section III contains an explanation of the methodologies used in the proposed model followed by algorithms used. Section IV is the operational analysis which contains the architecture and flowchart used in the system and the combined algorithm is also explained. Section V explains the metrics on which the system is evaluated and summarizes various operational results and findings. Section VI has the conclusion along with the future scope of the proposed solution.

2. Literature Review

A number of researchers are doing experiments to achieve better accuracy and reduce time for detecting the credit card frauds [6, 7, 8, 9, 10]. [11] have provided a review on the different credit card fraud detection practices. [12] proposed a hybrid of Bayesian Networks and Artificial Neural Networks as a technique to detect frauds in credit cards. Their paper consists of a discussion which states the speed of Bayesian networks was accelerated by ANN and after a short training period, they gave good results. [13] put forward a hybrid technique of artificial neural networks and decision trees. Firstly, the classification results obtained by the Decision tree and Multilayer perceptron generated a new dataset. This new dataset is then fed into Multilayer perceptron for the final classification of the data. High reliability is obtained by this model as a result of a very low false detection rate. [14] in their paper have devised a hybrid of K-means clustering combined with the Hidden Markov Model (HMM) and Multilayer Perceptron (MLP). The dataset is fed to K-means and then its output is given to HMM and MLP for their training which classifies the incoming transaction. As seen in the observations, the combination of "MLP with K-means clustering" gives more accurate results or higher accuracy. However, when used with 10-fold cross-validation the result is reversed. [15] in their paper Credit Card Fraud Detection Using Autoencoder Neural Network have proposed a de-noising autoencoder neural network (DEA) algorithm to handle the imbalanced nature of Credit Card datasets along

Techniques	Advantages	Disadvantages
Artificial Neural Network	• ANN has ability to learn from the past. It does not need to be reprogrammed. • ANN is capable of detecting the fraudulent activity during the transaction.	• High processing time in case of large neural networks. • Excessive training required. It is difficult to set up and operate. • Sensitivity to data format.
Bayesian Network	• High detection and processing speed. • High accuracy	• Excessive training needed. • Expensive
Support vector machines	• Deliver a unique solution, by choosing an appropriate generalization code • Robust	• Poor at processing large datasets • Expensive. It has a low speed of detection. • Medium accuracy. It lacks transparency
Fuzzy Logic	• Very fast in detection/accurate • High maintainability	• Low speed of detection • Highly expensive
Decision Tree	• High flexibility. Explainable • Easy to understand and implement • Can handle nonlinear data as well	• Cannot detect fraud during the transaction. • The algorithm is complex. Even a small change in data can distract the structure.
Hidden Markov	• Capable of detecting frauds at the time of the transaction • Reduces the false positive	• Cannot detect fraud in the initial few transactions. • Not scalable to large size datasets • Expensive
K-nearest Neighbour	• Predictive model is not required before classification	• The method accuracy depends on the measure of distance • Cannot detect fraud during the transaction.

with SMOTE (Synthetic Minority Oversampling Technique) and SoftMax function in neural network classification to model the system. [16] in their paper have trained their model using Artificial Neural Network and three different learning mechanisms Gradient Descent Adaptive Learning, Bayesian Regularization (BR) and LM algorithm, and found that BR gave the best results. Table 2 shows the various systems parallel to the proposed system.

Literature Reference	Approach	Finding	Remarks
[12]	Naïve Bayes and Artificial Neural Network	83.14% detection rate	ANNs are faster in detecting frauds through Bayesian network give better results with a shorter training period but they are comparatively slower.
[13]	Multi-Layer Perceptron Neural Networks and Decision Tree	99.89% detection rate	The proposed intrusion detection system will have a high detection rate and low false alarm rate. Hence the results produced are reliable.
[17]	Fuzzy Clustering and Neural Network	93.90% detection rate	By clubbing clustering technique with learning aids in effective detection of frauds.
[14]	K-means Clustering with Multilayer Perceptron Algorithm and Hidden Markov Model	80.5% detection rate	It is observed that Multilayer Perceptron combined with K - means clustering has outperformed the combination of Hidden Markov Model with K - means clustering.
[15]	Autoencoder Neural Network	84% detection rate	From the results, it can be concluded that the imbalance and noise in the minority class of the dataset could be removed using the autoencoder method.
[16]	Artificial Neural Networks	95.57% detection rate	There are various training techniques available to apply on the network. It was found that the Bayesian Regularisation Technique provided results with the best accuracy comparatively.
[8]	LSTM-RNN	99.58% detection rate	The desired features were extracted from two different datasets taken from the Kaggle repository using Principal Component Analysis (PCA) and then preprocessed using Arbitrary Assignment Method and Min–Max scalar algorithm for Normalization.

Table 1
Various Systems Parallel to the Proposed System

3. Materials and Methods

Neural Networks have the powerful capability of identifying patterns and the correlations and differences among those patterns. K-means Algorithm can detect outliers even in overlapping pattern set and Genetic Algorithm uses powerful concepts of evolution to generate an optimized system. Together they make a very fast, precise, and powerful fraud detection system. The advantages provided by these three techniques are leveraged in this tri-hybrid approach. In this section, the working of each of them is explained in detail.

3.1. K-means Clustering

Fuzzy Clustering also called soft clustering or K-means clustering is a clustering technique that separates data points into different clusters based on how much similar are they to each other and how much they differ from other data points [18]. In fuzzy clustering, one data point can belong to more than one cluster. Clusters are differentiated based on similarity measures such as intensity, connectivity, and distance. Fuzzy Clustering makes our system have higher accuracy and Lower False Alarm Rates. Given n data points $x_1, \ldots, x_n$ K-means aim to find k no. of centers $c_1, \ldots, c_k$ and assignments $q_1, \ldots., q_n$ of the data points to the centers such that sum of distances is minimized.

$$E(c_1......c_k, q_1....q_n) = \sum_{i=1}^{p} \sum_{q=1}^{k} n * ||x_i - (c_{qi})||^p * p$$

The first center c1 is selected at random from the data points x1,. . . ,xn, and then the distance between this center and all points xic1p is calculated. The second center c2 is then selected from the data points with their probability proportional to the distance. Using the minimum distance to the centers collected so far; repeat the procedure to obtain other centers. Each data point is assigned to the cluster from which it has a minimum distance. To calculate the distance of each data point from each of the centroids Euclidian Distance has been used. Other methods that can be used are Cosine Distance (cosine of the angle between the data points), Manhattan Distance (absolute difference between coordinates of the two data points), and Minkowski Distance (average or generalized distance). Once each data point has been assigned a cluster, we re-calculate the centroid as a mean of all its constituting data points. Then a cluster of all the data points is re-calculated and this process repeats till no data point shifts between clusters. The output of clustering is used to find out the transactions that are the outlier. An outlier is an object that is inconsistent concerning our data. They do not confer to the normal data and hence need to be evaluated. In the case of credit card fraud detection, we have a highly unbalanced nature of data i.e. the number of data of frauds in comparison to a genuine transaction is very minuscule. Hence finding transactions that are the outlier and sending only those to the neural network for classification not only makes our system better trained but also faster. Outlier detection is done by calculating the distance between each transaction in a cluster with the center of the cluster. All the transactions that fall above a threshold value that is calculated as the average of all the distances are assigned as the outlier.

3.2. Genetic Algorithm

A genetic algorithm is an evolutionary optimization technique [3]. It is a search algorithm based on the mechanics of natural selection and genetics. Genetic algorithms simulate the process of natural evolution, wherein each coming generation is made better and better by selecting the fittest individuals for reproduction. It searches for an optimal solution by feeding the candidate solutions into an algorithm, computing their fitness, and eliminating the worst-performing members. It then selects the rest of the members and produces offspring from them by performing some selection criteria such as crossover or mutation. These offspring become the new population which is again fed into the algorithm and the entire procedure is repeated till a stopping criterion is met, thus increasing the fitness of the system. The fitness function used in the model is the mean square error method also known as cross-validation score in genetic algorithm.

$$MSE = \frac{1}{M}\sum_{i=1}^{M}(y_{true} - y_{pred})^2$$

A Genetic Algorithm is used in the system to optimize four parameters which are:

1. Number of layers (depth),
2. Neurons in the layer (width),
3. Dense layer activation function, and
4. Network Optimizer.

The aforementioned parameters are chosen over the others because they are the most crucial parameters which play an important role in the correct classification. Hence, these parameters need to be made stronger than the others so that they have a greater say in output. If the number of layers is less than this could lead to weak computing and faulty processing while a large number of layers would cause the neural network to slow down. Hence, it is optimized using a genetic algorithm. Generally, how many neurons would comprise the input layer is decided by the number of variables in the input dataset which is being processed. The calculation of the number of neurons in the hidden layer is tricky but by the rule of the thumb, they should be smaller than or equal to the minimum number of neurons in the input or output layer approx. $\frac{2}{3}$ the size of the input layer plus the output layer. This is more accurately computed by feeding data on synapses to the genetic algorithm. A genetic algorithm helps the neural network by discarding the unnecessary as well as the insignificant neurons; thus, speeding up the learning. The dense layer activation function plays a major role in deciding which neuron will be activated which in turn has a significant impact on a correct classification. Network optimizers play a significant role in minimizing the loss function thereby contributing towards the error-free output. Hence, it needs to be chosen for optimization for genetic algorithms.

3.3. Neural Network

A neural network is a network that comprises several nodes (neurons) present in each layer. Each node of a layer is connected to every other node in the next layer and each edge connecting them has a weight assigned to them [19]. Activation of each node in the next layer depends on the sigmoidal function which computes data in each node in the previous layer by using the weights of the edges connecting them and some bias to give the activation of the next node. The nodes in layer one activate the nodes in layer two and further the process continues until finally a node in the final layer is activated, which is considered as the output. The total weight on a neuron Y with three input neurons will be

$$\frac{1}{(1+e)} - y$$

And activation of neuron Y will be given by

$$Y = W_i I_i = W_1 I1 + W_2 I_2 + W_3 I_3$$

Feed Forward Back Propagation Learning Algorithm In this fraud detection system, a five-layer feed-forward back propagation neural network is being used with supervised learning which is one of the most powerful learning algorithms [20]. The feed-forward will diagnose the transactions while the back propagation will calculate errors generated and accordingly correct the weights on the edges as they play the main role in the activation of nodes and thus in the output. Supervised learning means that we will be provided with the pair of input and output data values and will compare the generated output with the desired output to calculate the percentage of error. $Weight+ = Error * Input * Output(1 - Output)$ The feed-forward back propagation algorithm carries the data in one direction and does not allow loops either in a forward direction or in a backward direction.

4. Operational Analysis

The dataset applied in the present study is taken from the Kaggle repository, a subsidiary of Google LLC and is available at https://www.kaggle.com/mlg-ulb/creditcardfraud/home. It contains transactions made by credit cards in September 2013 by European cardholders over two days, having 492 frauds out of 284,807 transactions. The dataset is highly unbalanced with the positive class (frauds) accounting for 0.172% of all transactions. It has in total 31 features out of which 28 correspond to attributes of a customer like name, age, occupation, location, account balance, type of card, etc. For security purposes, they have been PCA transformed. The other three features are time, amount, and class.

4.1. System Architecture

Figure 1 depicts the architecture of our proposed model. The incoming credit card transaction is first matched against the fraud history database. Then with that information, it is passed to the customer profile analyst and then to the deviation analysts, who calculate the profile score and deviation score of that particular incoming transaction respectively. Then that transaction is fed as an input to the Fraud Detection System with those scores for future reference. This Fraud Detection System is a hybrid of the three most efficient machine learning techniques used for the detection of fraud namely K- Means Clustering, Genetic Algorithm, and Artificial Neural Network. If the transaction is detected to be normal then it follows the regular flow and proceeds to completion else if any anomaly is detected or risks are associated with that transaction then the alarm is raised. An alarm is raised in form of maybe OTP, e-mail, call, or text message to the customer as per the implementation. If the customer owns the transaction, then the transaction proceeds to completion. If the customer denies having initiated that transaction then the transaction is aborted then and there.

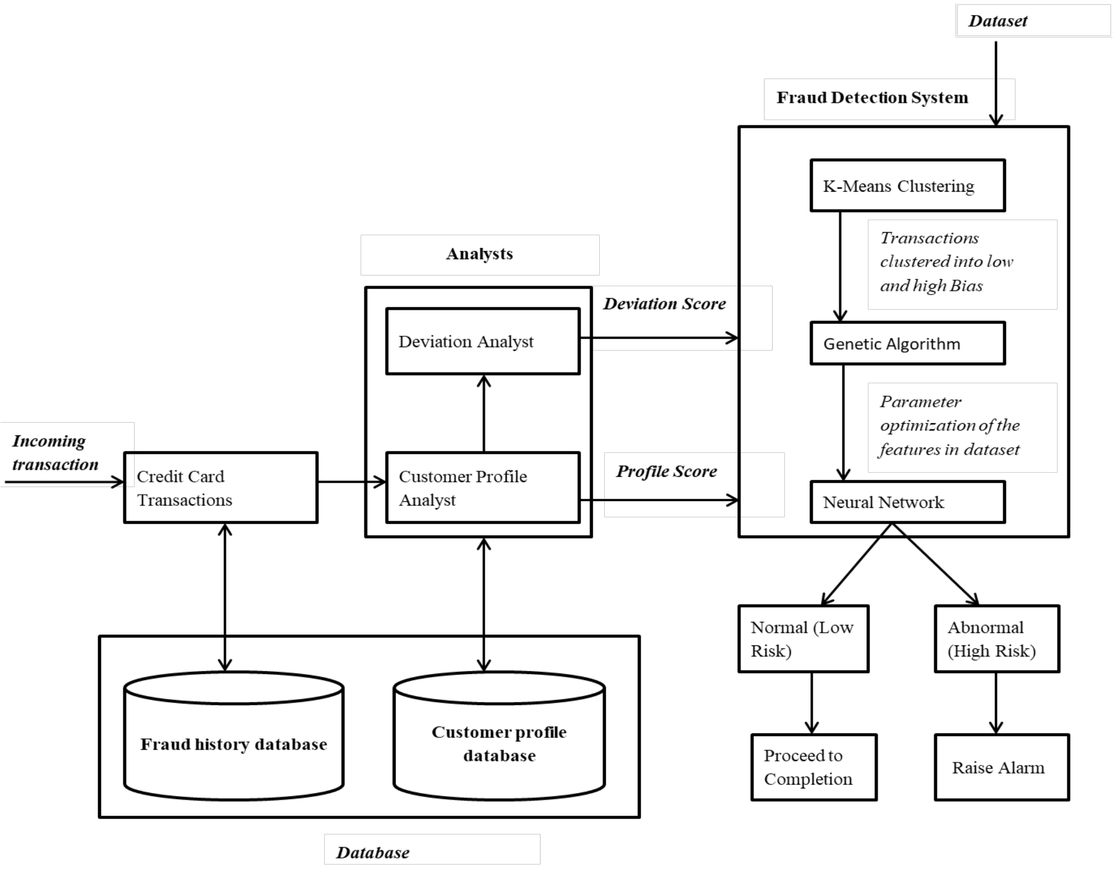


Figure 1: Architecture

4.2. Flowchart

Figure 2 depicts the flowchart of the working of the system. The dataset used is partitioned into two sets, Train Dataset and Test Dataset in the ratio 4:1. The training dataset then goes into the K-means clustering algorithm where clusters are formed and the transactions are then categorized to be of either low risk or high risk. If the transaction is of low risk then the bias send to the Neural network is lower otherwise it is higher. This bias of each transaction is input to a genetic algorithm which transfers this information to the neural network. A genetic algorithm (GA) also optimizes the parameters and the initial weight matrix that is input to a neural network for training. GA first randomly generates a generation, evaluates its fitness using fitness scores, and generates further generation using selection criteria of crossover and mutation. Once the stopping condition is met the dataset with the information of bias and parameter selection (weight matrix) is send as input to Neural Network to get trained by using the feed-forward back propagation algorithm. After training has been completed, a trained system is achieved on which test dataset runs to find the accuracy of the system.

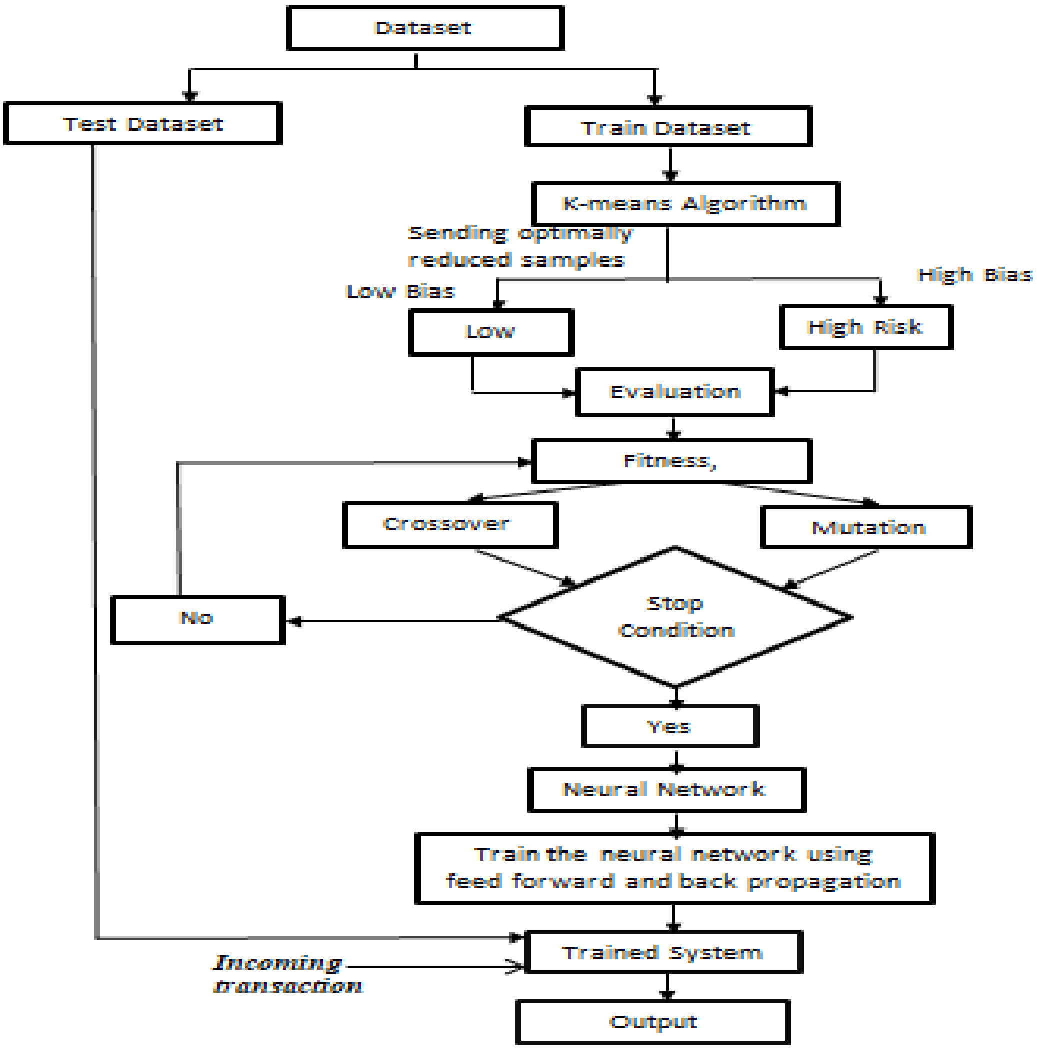


Figure 2: Flowchart

4.3. Algorithm Development

The dataset is firstly pre-processed by splitting into train dataset and test dataset. Split the dataset into 70:30 ratio for training and validation (testing) datasets i.e. 70% of the dataset rows are used as train dataset and 30% as the test dataset. The training dataset contains the rows and columns which would be used for training the network. The test data is then used against the trained network to measure the accuracy and other evaluation metrics. The last column “class” of the dataset is dropped as it is not needed during the training part. Removal of non-constant features is useful for better training of the model. After forming the clusters in the dataset, the distance of each point is calculated from its center, this is the K-means application. These data points are nothing but different column names in the dataset like be the location of the transaction, amount of transaction, etc. The closest cluster to each point is predicted iteratively until there is no shift between the clusters. This process optimally reduces the dataset for the genetic algorithm as the assignment of high and low bias has been included in each data point's information. The genetic algorithm now plays its part on these data points for the best feature selection. These features or parameters decide the structure of the neural network. The genetic algorithm forms the initial generation by creating a random combination of features. For example, for one neural network no. of layers might be 4 and for another, it might be 5. For each of these combinations of genes, the genetic algorithm performs mutation on some, crossover on some, and uses some (which have the highest score) directly for the next generation. As a result, the best suitable structure for the network is decided and then this network is formed using the Sequential() method which is a predefined method in the Sequential class of the Keras library. As the neural network is formed using the parameters given by the genetic algorithm, the test dataset comes into play. This data is then given to the neural network and the evaluation metrics are measured as the network shows which transaction is fraud and which is genuine to raise the alarm.

Here are the steps involved in the algorithm explained above. The input and the final output are mentioned before the algorithm begins. The input is a CSV file named creditcard.csv which contains the PCA reduced values of various transactions. The column names denote the features which will be clustered using the K-means clustering and the rows in the dataset are values corresponding to different transactions.

Algorithm: Trio-Hybrid Algorithm for Detection of Fraudulent Credit Card Transactions
Input: credicard.csv dataset
Output: Scalar consisting of loss and the values of the metrics.

1. Pre-process the dataset

- Split the dataset into train dataset D and the test dataset.
- Drop the column "class" of the dataset.
- Scale the train dataset D for standardization i.e. remove non-constant features.

2. Apply the techniques i.e. K-means clustering, genetic algorithm, and neural network.
2a. K-Means Clustering

- Form clusters and divide the sample data D among all clusters.
- Calculate the distance of each point in sample space from all the clusters using K-means algorithm.
- Predict the closest cluster to which each sample in D belongs.
- **If** the sample point is already closest to its own cluster
  - Stop

  Else
  - Move the sample point to the cluster it has least distance with.
- Repeat the above step until no sample shift between the clusters.

- The dataset has now been optimally reduced after the data points have been grouped accordingly into the clusters having similar data points.

2b. Genetic Algorithm

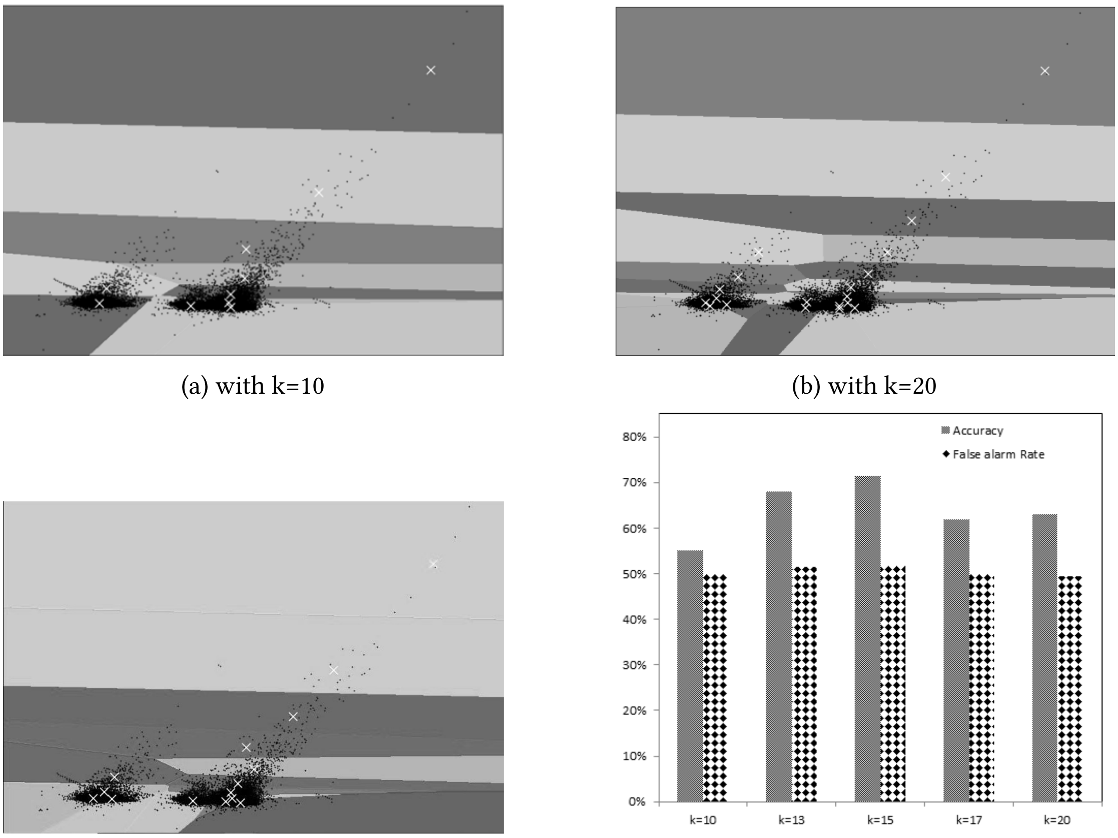
- The data points with their cluster information i.e. high risk or low risk are then used by the genetic algorithm to form a random combination of parameters such as number of layers in the network, number of neurons in the layer etc. to be used in the formation of most suitable neural network. This random combination forms the first generation for the genetic algorithm.
- Store this random combination of parameters into a variable GENES.
- Calculate the fitness score upon each of the generated combination or GENES by mean squared error fitness () function: $MSE = \frac{1}{M}\sum_{i=1}^{M}(y_{true} - y_{pred})^2$ Lesser the score, fitter the GENES.
- Sort the population of these combinations into increasing order of fitness score.
- **If** fitness score is found as 0
  - Stop

  Else
  - 10% of the current fittest population is taken to be included in the next generation
  - 50% are mutated to produce offspring i.e. take two combinations and permute the combination of parameters i.e. perform crossover
  - For the rest of the GENES, insert a random parameter into their combination i.e. perform mutation.
  - Repeat the above steps until a best score is achieved for a set of parameters.
- The most suitable parameters for the neural network are thus obtained after best feature selection by the genetic algorithm. grouped accordingly into the clusters having similar data points.

2c. Artificial Neural Network

- Create object of Sequential()
- Use this object to add input and output layers to the model (making of the network).
- Fit the data into the network i.e. train the network using Epoch=200, batch size=500
- This network is then tested against the test dataset by considering the output produced which is a scalar consisting of loss value and metrics value to indicate the fraud and genuine transactions.

End



(a) with k=10

(b) with k=20

(c) with k=15

(d) Accuracy and FAR at different values

Figure 3: K-means (a) with k=10 (b) with k=20 (c) with k=15 (d) Accuracy and FAR at different values

5. Experimental Analysis

The results obtained by experiments and the metrics on which they are evaluated are explained below in the two subsections. The results are analyzed and discussed through graphs and figures obtained through experiments on different parametric values. No proposal can be modelled into a system without some experiments to support it. The results and outputs included have been produced by this system under various inputs and parameters. To get the optimal value of K in the K-means algorithm we varied the number of clusters from 10 to 20. Figure 3a shows clustering when K=10 and Figure 3b shows clustering when K=20. It depicts that with an increase in the number of clusters there is a higher chance of forming better clusters. While running the K-means clustering algorithm on different K values it was observed that the best accuracy is given when K was 15. Figure 3c shows the output produced by the K-means clustering algorithm when K=15. In figure 3d the accuracy of the algorithm at different K values is shown. Fig. 4(a) and Fig. 4(b) shows the cross-validation score of best genes and average genes in each generation at seed value 108 and 10000 respectively. The relationship between initial seed values given to genetic algorithm and mean square error of cross-validation score generated

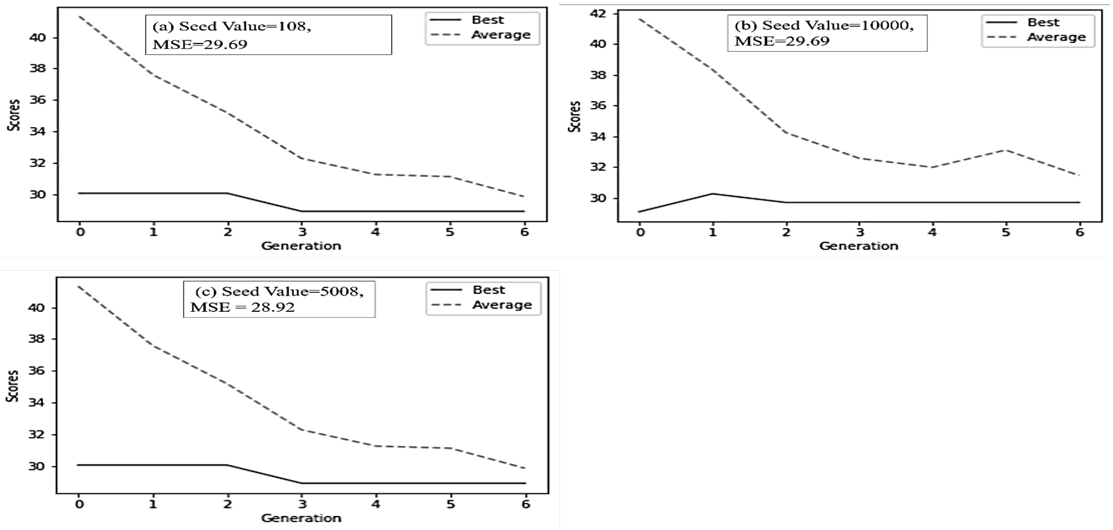
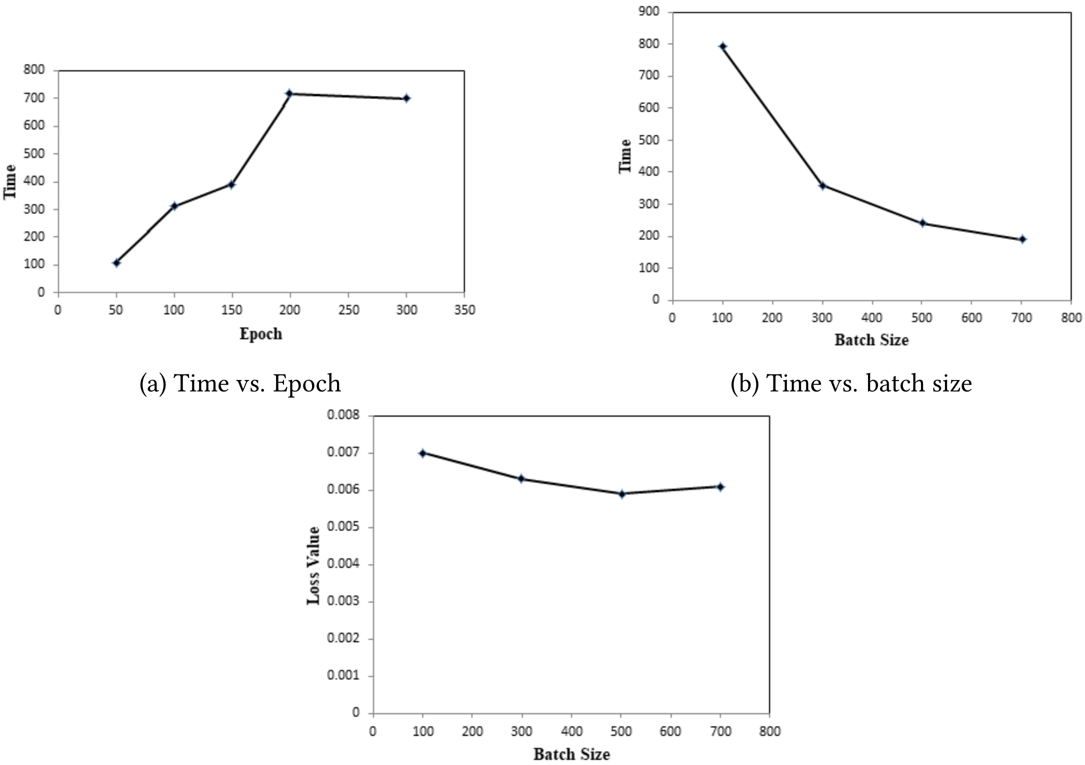


Figure 4: Graphs showing different mean square error after feature selection

for that seed has been depicted. It can be observed from Fig.4(c), that when the seed is around 5000 the mean square error is minimum. It increases both decreasing the seed value as well as increasing the seed value. It is shown that before feature selection of the most important features, the mean square error was 37.13% which was reduced to 28.92%. Although the number of generations taken for the average and best features in graphs Fig.4(a) and Fig.4(b) to reduce the fitness score (lesser the score, better the generation) is a little lesser compared to Fig.4(c) i.e. on a seed value of 5008, it is compensated by the reduced mean square error. This score hugely depends on the initial seed value. Fig.5a shows the time required by the neural network to train at different Epoch values. Fig.5b depicts that when we increase the batch size the time taken to train the neural network gets reduced. With the increase in batch size, the loss value decreases up to a certain batch size which is shown in Fig.5c. We get the least loss value at a batch size equal to 500 and after that increasing the batch size though decrease the time taken by the neural network to get trained but it also increases the loss value. After getting a trained neural network we run the test data set on the system and the observations made on the output are mentioned below. Accuracy = 99.94% Loss Value = 0.561%

6. Conclusion

The system described here is faster than the present systems as the training time of neural networks is reduced by the use of the k-means algorithm and the efficiency is increased by the use genetic algorithm. The genetic algorithm helped in the best parameter selection and removed the redundant parameters. 100 epochs are ideal for this fraction of the dataset since lesser than this will cause under-fitting of the system and if more epochs are used then the system will be over-trained for that particular dataset. Henceforth, the results indicate that the hybrid of these three techniques gave a faster and optimized system which is the need of the



(a) Time vs. Epoch

(b) Time vs. batch size

(c) Loss value vs. Batch size

Figure 5: (a) Time vs. Epoch (b) Time vs. batch size (c) Loss value vs. Batch size

present global scenario.

The loophole in the existing systems is that they aren't able to adapt themselves quickly in the changing environment which is compensated by the use of k-means clustering The only setback of the aforementioned system as of now is the cost of its implementation since its complex to implement because of its hybrid nature. Fraud detection is a field that will never be dormant as there are always new strategies that can be found to commit fraud. Various other hybrid techniques could be experimented as future scope for better results.

References

[1] M. Zanin, M. Romance, S. Moral, R. Criado, Credit card fraud detection through parenclitic network analysis, Complexity 2018 (2018).

[2] M. S. Kumar, V. Soundarya, S. Kavitha, E. Keerthika, E. Aswini, Credit card fraud detection using random forest algorithm, 2019 3rd International Conference on Computing and Communications Technologies (ICCCT) (2019) 149–153.

[3] S. Bhattacharyya, S. Jha, K. Tharakunnel, J. C. Westland, Data mining for credit card fraud: A comparative study, Decision support systems 50 (2011) 602–613.

[4] R. Patidar, L. Sharma, et al., Credit card fraud detection using neural network, International Journal of Soft Computing and Engineering (IJSCE) 1 (2011).

[5] M. Zareapoor, K. Seeja, M. A. Alam, Analysis on credit card fraud detection techniques: based on certain design criteria, International journal of computer applications 52 (2012).

[6] M. Seera, C. P. Lim, A. Kumar, L. Dhamotharan, K. H. Tan, An intelligent payment card fraud detection system, Annals of operations research (2021) 1–23.

[7] N. K. Trivedi, S. Simaiya, U. K. Lilhore, S. K. Sharma, An efficient credit card fraud detection model based on machine learning methods, International Journal of Advanced Science and Technology 29 (2020) 3414–3424.

[8] O. Owolafe, O. B. Ogunrinde, A. F.-B. Thompson, A long short term memory model for credit card fraud detection, in: Artificial Intelligence for Cyber Security: Methods, Issues and Possible Horizons or Opportunities, Springer, 2021, pp. 369–391.

[9] N. Boutaher, A. Elomri, N. Abghour, K. Moussaid, M. Rida, A review of credit card fraud detection using machine learning techniques, in: 2020 5th International Conference on Cloud Computing and Artificial Intelligence: Technologies and Applications (CloudTech), IEEE, 2020, pp. 1–5.

[10] R. Dash, R. Rautray, R. Dash, A legendre neural network for credit card fraud detection, in: Intelligent and Cloud Computing, Springer, 2021, pp. 411–418.

[11] Y. Jain, N. Tiwari, S. Dubey, S. Jain, A comparative analysis of various credit card fraud detection techniques, Int J Recent Technol Eng 7 (2019) 402–407.

[12] S. Maes, K. Tuyls, B. Vanschoenwinkel, B. Manderick, Credit card fraud detection using bayesian and neural networks, in: Proceedings of the 1st international naiso congress on neuro fuzzy technologies, volume 261, 2002, p. 270.

[13] J. Esmaily, R. Moradinezhad, J. Ghasemi, Intrusion detection system based on multi-layer perceptron neural networks and decision tree, in: 2015 7th Conference on Information and Knowledge Technology (IKT), IEEE, 2015, pp. 1–5.

[14] S. G. Fashoto, O. Owolabi, O. Adeleye, J. Wandera, Hybrid methods for credit card fraud detection using k-means clustering with hidden markov model and multilayer perceptron algorithm (2016).

[15] J. Zou, J. Zhang, P. Jiang, Credit card fraud detection using autoencoder neural network, arXiv preprint arXiv:1908.11553 (2019).

[16] C. Mishra, B. Gupta, R. Singh, Credit card fraud identification using artificial neural networks, International Journal of Computer Systems 4 (2017) 151–159.

[17] T. K. Behera, S. Panigrahi, Credit card fraud detection: a hybrid approach using fuzzy clustering & neural network, in: 2015 second international conference on advances in computing and communication engineering, IEEE, 2015, pp. 494–499.

[18] P. Chougule, A. Thakare, P. Kale, M. Gole, P. Nanekar, Genetic k-means algorithm for credit card fraud detection, International Journal of Computer Science and Information Technologies (IJCSIT) 6 (2015) 1724–1727.

[19] T. Razooqi, P. Khurana, K. Raahemifar, A. Abhari, Credit card fraud detection using fuzzy logic and neural network, in: Proceedings of the 19th Communications & Networking Symposium, 2016, pp. 1–5.

[20] F. Amato, N. Mazzocca, F. Moscato, E. Vivenzio, Multilayer perceptron: an intelligent model for classification and intrusion detection, in: 2017 31st International Conference on

Advanced Information Networking and Applications Workshops (WAINA), IEEE, 2017, pp. 686–691.