

Multilayer perceptron artificial neural networks-based model for credit card fraud detection

Bassam Kasasbeh, Balqees Aldabaybah, Hadeel Ahmad

Department of Computer Science, Faculty of Information Technology, Applied Science Private University, Amman, Jordan

Article Info

Article history:

Received Sep 17, 2021

Revised Feb 2, 2022

Accepted Feb 15, 2022

Keywords:

Artificial neural networks
Credit card fraud
Machine learning
Multilayer perceptron online transaction

ABSTRACT

Nowadays, credit card fraud has emerged as a major problem. People are becoming increasingly using credit cards to pay for their transactions, it has become more popular and essential in our lives. Fraudsters are developing new strategies and techniques over time, and it is not easy for humans to manually check out all transactions. The cost of fraudulent transactions is significant and without prevention mechanisms it is rising. Finding the best methodology to detect fraudulent transactions is a crucial asset to the industry to reduce the fraud financial loss. Artificial neural networks (ANN) technique is considered as one of the effective techniques that has proved its efficiency in detecting credit card fraud transactions with high precision and minimum cost. In this paper, we propose a multilayer perceptron (MLP) ANN-based model solution to improve the accuracy of the detection process. The performance of the methodology is measured based on the precision, sensitivity, specificity, accuracy, F-measure, area under curve (AUC) and root mean square error (RMSE). Moreover, we illustrate the performance results of these measures with a descriptive analysis. Experimental results have shown that the proposed ANN-based model is efficient and does improve the accuracy of the detection of fraudulent transactions.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Bassam Kasasbeh

Department of Computer Science, Faculty of Information Technology, Applied Science Private University
P.O.BOX 166 Amman 11931, Jordan

Email: b_kasasbeh@asu.edu.jo

1. INTRODUCTION

Detecting credit card fraud is becoming more and more important in our lives. The need to pay over internet via credit cards are increasingly a demand. Although the number of credit card fraud patterns may form a small percentage of transactions, but the cost may be huge [1]. By preventing fraudulent transactions, we can save a significant amount of money that would traditionally be lost.

Detection is a data mining technique that relies on analyzing past records in order to find the desired pattern. In case of fraud detection, the desired pattern is the ability to distinguish between what is fraud and what is not. In financial payments, the credit card fraud transaction can be defined as the unauthorized usage of credit card data to conduct a financial transaction or remove balance from the account [2]-[4]. Generally, credit card fraud can be conducted by fraudulent physically (actual steal), or virtually. Because virtual fraud is more likely to happen, and riskier, as the user will not know about it till the financial institution blocks the transaction or the user reports abnormal behavior, means it can be undetected for long time. Therefore, fraud detection is considered as an online problem where the presence of a fraud detection system for every financial institution is a must [5].

Credit card fraud detection has some difficulties. First, due to the time and cost, it is not easy for humans to check all transactions and manually figure out incorrect patterns. The huge number of transactions

makes it uneasy for a human analyst to detect fraudulent patterns [6]. In real world, the massive transactions are scanned by automated tools, analyzed by classifiers who in turn find and alert the suspicious ones, alerts are then inspected by professional investigators [6]. If we put fraud detection performance in our consideration, then the massive transactions cannot be verified by investigators due to time and cost challenges [7]. Second, due to some reasons the distribution of the dataset changes over times, for example, Cardholders may change their purchasing behavior over seasons, and fraudsters strategies and techniques evolve over time, this is called concept drift, which refers to when the dataset shift [8]. In the presence of concept drift, it is difficult to decide whether a transaction is fraudulent or original based on previous data [9]. In the presence of this issue, the trained detection models on past records will be less efficient of detecting fraudulent patterns when new records arrive. Thus, in such situation, we need to guarantee a high accuracy of the system. Third, it is very challenging to find the best strategy to detect fraudulent patterns since the data are scarcely available due to confidential issues and typically, they are unbalanced [6]. Finding the right strategies is a crucial asset to the industry to enhance the fraud detection process.

Taking all the difficulties into consideration, it is important to keep a balance between the need of reducing the number of fraudulent transactions to be detected by humans and maintaining a high accuracy of the system when the dataset is shifted. We aim to find the best methodology in artificial neural networks (ANN) to implement a credit card detection solution that provides high precision and that is critical to saving time and money. In this paper we focus on measuring the solution correctness by its ability to detect frauds correctly rather than classifying a normal one as a fraud. Neural network is one of the machine learning classification techniques that has proved its efficiency in pattern recognition areas (e.g., detection problems), and its fast convergence [10].

In real-world, the process of collecting real credit card dataset is difficult since there are some restraints due to privacy and confidential issues. We considered a real dataset made by European cardholders' transaction as the source domain. The dataset contains transactions collected from European cardholders in September 2013 and it is highly unbalanced. The dataset is unbalanced due to that the number of fraudulent transactions is much smaller than the valid transactions [11]. The rest of the paper is organized as: Section 1 describes the related work, section 2 explains our methodology, section 3 demonstrates our experiments, and section 4 concludes the paper.

2. RELATED WORK

The problem of credit card fraud detection has been early investigated in the literature. Many different techniques and different approaches have been used to handle the issues of imbalanced dataset, concept drift, and verification/feedback latency as well as other issues. Credit card fraud detection is a hot research area in which there is always a place for improvement due to the high change rate in customer behaviors when using their cards, as well as the fraudster actions change over time (concept drift). Few works have explored this issue, proposed a real-time framework using SOM [12]. The effectiveness of incremental learning model in which when new transactions are added to the dataset, the model is updated to adapt to the presence of the new instances of data [6].

As in this paper we choose to handle the problem using artificial neural network (ANN) we will take the chance to further explore the applied ANN in the literature for fraud detection. One of the early systems of credit card fraud detection suggested by Aleskerov *et al.* [13] was CardWatch system. A feed forward neural network architecture was proposed as part of the system, in which it was trained using past records of customer to be able to detect possible frauds. Experiments showed that fraud detection rate was 85%. On the other hand Brause *et al.* [14] used ANN to investigate the impact of training the model with symbolic and/or analog features. The main goal was to help in minimizing the number of false alarms to be recognized by the financial institution as fraud or not. Their model was evaluated based on the confidence value. Similarly, Guo and Li [15] used confidence-based ANN to be able to detect fraudulent transactions. Their method basically relied on a confidence threshold value to classify if a transaction is a fraud or normal. Authors suggested ROC analysis technique to ensure the value of threshold is reasonable.

Others have used the ANN combined with other techniques to enhance and improve the detection results. According to Bahera and Panigrahi [16] suggested a methodology to reduce the misclassification rate using a feed forward back propagation (FFNNBP) neural network in a later phase of the system combined with fuzzy logic. The aim of ANN was to strengthen the initial observation of a transaction that has been classified as fraud by fuzzy algorithm in the early phase. This ensured if the suspicious transactions are actually fraud or not based on similarity with trained patterns. Results showed that 93.90% of transactions were correctly classified. Similarly, Wang *et al.* [17] used three layers feed forward back propagation neural network (BPNN) with whale optimization algorithm (WOA) proposed as (WOA-BP) where the role of WOA was optimizing the weights in BPNN to enhance the convergence speed of the training. According to the

evaluation with similar algorithms from the literature and based on real dataset, WOA-BP shows the smallest mean squared error, and the fastest convergence rate among other similar strategies in the literature.

The acquiring of real-world credit card dataset is difficult for mainly security and privacy issues. Despite of the lack of data because of sensitivity, few works in the literature have explored the effectiveness of using ANN models experimented and evaluated on real datasets. For example, Fu *et al.* [18] proposed a convolutional neural network (CNN) to avoid the model overfitting. The experiment was conducted on real transactions given by a commercial bank with around 26 M transactions, that has almost 4K as frauds. Authors had split the data for training, validation and testing based on transactions timeline.

Others have investigated different types of deep learning networks. According to Roy *et al.* [19] explored the results of training ANN, long short-term memory (LSTM) and gated recurrent unit (GRU) for the credit card fraud detection problem. The experiment was conducted on real dataset from a financial institution, where results has shown that GRU have outperformed other models in term of accuracy.

The most related work to our proposed solution was the one found in [20]. In this work authors have proposed ANN backpropagation algorithm. The experiment was conducted on the same dataset as the one we are using. In addition, authors have suggested under sampling as the sampling technique to handle the imbalanced dataset issue. Performance metrics have shown that 94.2% accuracy on imbalanced dataset as the false positive (FP) was 1.3%. In comparison to our proposed solution in this paper, we are investigating the correctness of building ANN classification model without manipulating the imbalance dataset in which we choose the best model based on an iterative approach.

On the other hand, many researchers have studied the application of different other machine learning techniques on the fraud detection problem. This is because the fraud detection problem is rich of issues that can be enhanced and improved by metaheuristic and fuzzy concepts. Researchers have used Bayesian network algorithm, Behera and Panigrahi [16] proposed a fraud detection approach that goes in three steps. The main supervised learning is at the third step where a neural network learns among the suspicious transactions to determine if it was actually a fraud or not. Maes *et al.* [21] investigated the application of Bayesian network in fraud detection with ANN to solve the uncertainty issue. In addition, Panigrahi *et al.* [22] proposed a novel approach that consists of 4 main phases where not only records from past were used to detect fraud, but also current transactions. Bayesian network was used as last check in the proposed model. Based on the Bayesian learning, the model was able to detect fraud transaction out from the suspicious ones.

Moreover, Srivastava *et al.* [23] showed how credit card fraud can be detected with high coverage and low false alarm rate using hidden Markov model (HMM). He stated how HMM can decrease the false positive transactions. Many other works had used different techniques for the detection problem, for example Bhattacharyya *et al.* [24] investigated the application of support vector machine (SVM) and random forest in binary classification problem as fraud detection. Where SVM was able to map high-dimensional feature space of inputs without adding computational complexity and it performed well with under sampling technique to handle the skewed data. Similarly, Sahin and Duman [25] proposed SVM and RF for credit card detection where RF outperforms the SVM for relatively small amount of transactions. Other techniques include decision tree [25], [26], genetic algorithms [27], logistic regression [28], and associated rules [29]. Table 1 summarizes the most related work found in literature that applied ANN to investigate the problem of credit card fraud detection.

Table 1. Three-line representation

Work/system	Main goal/motivation	ANN architecture	Measurements	Main Results
CARDWATCH	Used ANN as part of the whole detection system	Feed forward ANN with 3 layers	(RMSE)=0.16 set to detect fraud transactions	85% fraud detection rate
Fuzzy logic and neural network	Minimize the misclassifications by fraud detection system	FFBPNN, used at later phase	TP, FP and ROC	93.90% (TP) correctly classified transactions
Deep learning	Investigate different NN	ANN, ANN, LSTM and GRU	Accuracy	GRU outperform others
Neural Network	Decrease number of fraud transactions that are misclassified (FP)	ANN BP with scaled conjugate gradient	Accuracy and FP	94.2% accuracy (FP) 1.3%

3. METHODOLOGY

Fraud detection is considered as an online problem and requires reliable and flexible methods to detect the presence of credit card fraud in real-time. Nowadays, deep learning is the most powerful and interesting machine learning technique, and artificial neural network (ANN) is an efficient way to improve

the performances of fraud detection systems. The most favorable point associated with ANN is comprehensibility, learning from example, tolerance to noisy, and parallelism. In this Section, Subsection 2.1 describes the dataset that was used to assess the generalization of the ANN models.

3.1. Dataset description

Obtaining credit card fraud datasets is very difficult because banks do not publish any data related to customer's transactions. In this paper, we used a publicly available dataset, the Credit-card dataset [30]. The Credit card fraud detections dataset was obtained from a machine learning group (MLG) and artificial intelligence (AI) workbench public platform as a part of the Université Libre de Bruxelles (ULB). This dataset was gathered in September 2013 from MasterCard transactions of European cardholders. The dataset contains transactions created in a record period in just two days. Wherever there are a total of 285,607 transactions in total, with 0.172% of them being fraud cases. This dataset contains 31 features, 28 of which have been turned into numerical input values using principal component analysis (PCA) due to a confidentiality concern. These features are named as V1, V2,... , and V28. According to [31], the feature contains general information such as gender, marital status, age in months, Housing, Job, Telephone, Foreign worker, and identification (ID). Other banking information includes credit limit, past month bills, past month payments, account status, Wage assignments, other existing credits, credit history, purpose, saving account, credit amount, debtors, property, number of existing credits, credit card number, and personal identification number (PIN). time, amount, and class are the other three labeled variables that do not bind with principal component analysis (PCA). The time variable records the amount of time that has passed between each transaction and the first transaction in the dataset in seconds. The transaction amount is stored in the amount variable. Class variable maps the output 1 in case of fraud detection and 0 for normal transactions. The dataset statistics are summarized in Table 2.

Table 2. Dataset summary statistics

Number of Features		Number of Instances	
	31		284807
28 unlabeled features obtained after PCA	3 labeled features (Time, Amount, Class)	492 for Fraudulent cases.	284315 for Normal cases

3.2. Artificial neural network

Machine learning techniques can extract useful patterns and information from large datasets and is used for decision-making tasks and to evaluate future events probabilities. The insights derived from Machine learning are used for fraud detection, marketing, and scientific discovery [32], [33]. In this paper, we used the ANN technique with multi-layer perceptron (MLP) to generate a model to detect credit card fraudulent transactions. ANN is a mathematical model based on the emulation of the biological neural system. This technique consists of three main stages: multiplication, addition, and activation. The value of each artificial neural is multiplied by individual weights. On the middle side of the ANN is the total function that includes all the inputs' weight. At the end of the ANN is the total input that has been weighted and already went through the activation phase that is also called the transfer function [34].

MLP is a feed-forward ANN made up of a group of neurons linked together by linking weights. MLP converts a set of inputs into the desired outputs. Figure 1 shows the construction of MLP, which is made up of three basic parts: an input layer, a hidden layer, and an output layer. The input layer gets the data, which it then transmits to the first hidden layer, which forwards it until it reaches the output layer [34], [35].

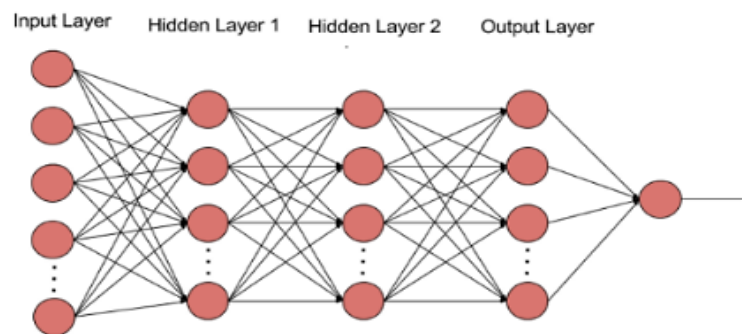


Figure 1. MLP

Each layer is made up of a certain number of neurons. Weight and bias are used to connect neurons between layers. The following equation yields the output (O_j) of each artificial neuron j in the hidden layer:

$$O_j = f(\sum_{i=1}^n w_i x_i + b) \quad (1)$$

where n is the number of neurons in the previous layer, w is the weight, x is the input value, b is the bias, and f the sigmoid activation function:

$$f(x) = \frac{1}{1+e^{-x}} \quad (2)$$

3.3. Model building

In this paper, we aim to find the best ANN model for fraud detection system. Due this end, variations of the number of neurons and the hidden layers have been tested. The number of input data and output units of multiple layers of neurons, the complexity of the classification problem, and the number of training cases to be learned all affect the best model [36]. Therefore, an iterative process is used in this paper to determine the best model. A 10-fold cross validation approach is developed in the iterative process to access all feasible models. Algorithm 1 describes the steps towards finding the best ANN model.

Algorithm 1. Find-best-ann-model

```

1: Procedure findBest(D,C)
2:   ▶ Input: D (array of n datasets), C (Configuration set of NN)
3:   ↳ C CONTAINS (L,M,N,V,S,E,H)
4:   ▶ Output B = B_1, B_2, ..., B_n
5:   ↳ B_i is the best model of dataset i
6:   For d_j ∈ D do
7:     N_c ← number of neurons in the output class
8:     For NHL ← 1 till 3 do
9:       IF NHL ← 1 then
10:        N_p ← number of neurons in the input layer
11:       Else
12:        N_p ← number of neurons in the previous layer
13:       End IF
14:       #generate variations of the same hidden layer
15:       N ← [(N)_p + N_c] / 2
16:       V ← [N-1, N, N+1] #three hidden layers with different number of N
17:       Foreach element in V do
18:         T ← Build NN(V_i, C)
19:         Trained ← Train(T) #return F-score
20:         B_i ← if(Trained is better than previous models)
21:       End For
22:     End For
23:   End For
24: End Procedure (return B)

```

According to Algorithm 1, the procedure starts with defining the model configurations to ensure the fair of comparisons. As first step, we start by trying all the variations of neurons within one, two, and three hidden layers (for in line 7) as experiments show that adding more hidden layers gives no better results than these layers. The process of selecting the number of neurons in the current hidden layers is done according to the equation in line 13. After considering three variations of neurons number for the hidden layer, it is now the time of finding the best result given the number of hidden layers, number of neurons in each hidden layer, and the required configurations (for loop line 15-19). Line 17 fined the best model in term of the F-measure since other measurements terms may be misleading when dealing with imbalanced dataset (see section 2.4).

The following are the properties of all ANN models trained in this experiment:

- A multilayer perceptron algorithm (MLP) neural network artificial was used.
- The number of neurons (nodes) in the input layer was 29, and there were two output neurons (nodes).
- For training, the backpropagation (BP) algorithm was used.
- The sigmoid activation function was used to test each model by using (2).

3.4. Performance evaluation

The metrics considered in the performance evaluation of the models in this paper are given as:

- The confusion matrix is used to evaluate the result, and precision, recall, and accuracy are calculated. It has two types of classes: actual and predicted.

- Accuracy: The ratio of total predicted transactions correctly classified.
- Sensitivity (Recall): is a metric that indicates how often a test accurately generates a positive result for observations that have the condition being tested for.
- Specificity: is a percentage of correctly classified abnormal samples. Higher the specificity, better the model.
- Precision: indicates how many of the cases that were correctly predicted turned out to be positive.
- F-Measure: The weighted harmonic means of the two fractions precision and recall of the test, which is a measure of a test's accuracy. The outcome is a number between 0.0 and 1.0, with 0.0 being the worst F-measure and 1.0 representing the best F-measure.
- Area under curve (AUC): represents the degree of separability. It indicates how well the model can distinguish between classes.
- Root mean square error (RMSE): calculates the differences between values predicted by a hypothetical model and the observed values. It measures the accuracy of the projected model's fit to the actual data.

4. RESULTS AND DISCUSSION

In order to apply the classification, we applied Weka 3.8.5. Experiments were conducted on an Intel Core i5 with 8.0 GB of RAM and 2.40 GHz processor with Windows 10 64-bits operating system. We used 10-Fold cross-validation (CV) to partition the training dataset into 10 equal portions and extract varied results. This method uses 9 of the 10 elements to train a neural network and the remaining part to evaluate it. The identical procedure is followed for all ten parts, with the exception that the test set is determined using a sliding window and the remaining parts are used to train the neural network. Following that, the results are collated, and the averages of all folds are calculated. The 10-Fold CV main aspect is that it uses all of the records in the dataset alternately for training and testing.

The best ANN model for the fraud detection system was determined through an experimental study. Accuracy is an important metric to consider but it does not always give the full picture. Sometimes when using only accuracy to find the best model can be misleading [37], [38], especially with an unbalanced dataset. The dataset used is very unbalanced which is biased towards the category of non-fraud cases. For such problems, classifiers should be evaluated with additional measures.

The Authors in Boughorbel *et al.* [39] showed that using accuracy as a performance measure in unbalanced datasets was insufficient. In dealing with the imbalanced data, the authors found that the F-measure and area under curve (AUC) produced higher consistency than accuracy. For that, we used the 10-fold cross validation method to run many experiments to find the best ANN model with the best classification F-measure.

4.1. Results analysis with one hidden layer

Several network structures were investigated to discover the optimal ANN model to use. Because there are 29 neurons in the input layer and two neurons in the output layer. The number of neurons in the first hidden layer will be 15.5, according to Algorithm 1. Therefore, the following models will be formed in the first hidden layer: 29-15-2, 29-14-2, and 29-16-2. Table 3 presents the confusion matrix for the best model with one hidden layer (29-14-2). This model represents 29 input variables, one hidden layer of 14 neurons and two output layers. The F-measure for fraud class in this model is 84.757%. All the remaining models produced slightly low classification F-measure 84.021% for 29-16-2 model, and 82.366% for 29-14-2 model. The root mean square error (RMSE) of this model by using (9) is 0.0218. Table 3 presents the confusion matrix and the result summary for this model.

Table 3. Summary results best model with one hidden layer

		Predicted		F-measure %	Precision %	Sensitivity %	Specificity %
		Normal	Fraud				
Original	Normal	284274	41	99.975%	99.986%	99.965%	79.675%
	Fraud	100	392	84.757%	79.675%	90.531%	99.986%

4.2. Results analysis with two hidden layers

In the first hidden layer, we obtained three different numbers of neurons; 14, 15 and 16. Therefore, by applying step 2 from our methodology, the number of neurons in the second layer will be as the following: i) When the first hidden layer has 14 neurons, the second hidden layer contains 14/2 neurons, which is 7 neurons. The network models will be 29-14-7-2, 29-14-6-2, and 29-14-8-2. ii) When the first hidden layer

has 15 neurons, the second hidden layer contains 15/2 neurons, which is 7.5 neurons. The network models will be 29-15-7-2, 29-15-6-2, and 29-15-8-2. iii) When the first hidden layer has 16 neurons, the second hidden layer contains 16/2 neurons, which is 8 neurons. The network models will be 29-16-8-2, 29-16-7-2, and 29-16-9-2.

After we tested 9 different models with two hidden layers, we found that the 29-15-8-2 model has the best F-measure value for fraud class 85.129% with an error 0.0214. Also, this model has the best precision, sensitivity, and specificity. Table 4 presents the confusion matrix and the result summary for this model.

Table 4. Summary results the best model with two hidden layers

		Predicted		F-measure %	Precision %	Sensitivity %	Specificity %
		Normal	Fraud				
Original	Normal	284274	41	99.976%	99.986%	99.966%	99.986%
	Fraud	97	395	85.129%	80.285%	90.596%	80.285%

4.3. Results analysis with three hidden layers

We obtained 9 different models when the number of hidden layers was two so that when a third hidden layer is added, 27 different models were tested. Table 5 shows the confusion matrix and the summary result for the best classification F-measure for fraud class is 83.193% is obtained when the network model is 29-15-8-4-2. When we trained the dataset with four and five hidden layers, the same results are always obtained in all possibilities with different options such as changing the learning rate. As we note from Table 6 that all 492 fraud cases were classified as normal cases, due to the large gap between the number of frauds and normal cases in the dataset.

Table 5. Summary results the best model with three hidden layers

		Predicted		F-measure %	Precision %	Sensitivity %	Specificity %
		Normal	Fraud				
Original	Normal	284246	69	99.971%	99.976%	99.966%	99.976%
	Fraud	98	394	82.513%	80.081%	85.097%	80.081%

Table 6. Summary results the best model with three hidden layers

		Predicted		F-measure %	Precision %	Sensitivity %	Specificity %
		Normal	Fraud				
Original	Normal	284315	0	99.914%	100.00%	99.827%	100.00%
	Fraud	492	0	NaN	0.000%	NaN	0.000%

4.4. Results analysis with four hidden layers

We started building the ANN models with one hidden layer and ended with 5 hidden layers. We can notice that the best model was obtained by using two hidden layers; the model is 29-15-8-2 and obtained 99.950% classification F-measure. Using one hidden layer achieves good results, but when another hidden layer has been added, the best classification F-measure has slightly risen from 84.757% to 85.129%. Adding three hidden layers lowered the F-measure and did not improve the results.

These results can be explained as: If you have too few hidden layers, you will get high training error and high generalization error due to under fitting. This part deals with the summary results collected during experiments. Table 7 compares results between the best ANN model in the one hidden layer, two hidden layers, and three hidden layers. Parameters chosen for comparison of results are F-measure, precision, sensitivity, specificity, accuracy, AUC, and RMSE. We notice that in all cases the results are remarkably close to each other, and this is expected due to the structure of the ANN. This is because that the ANN works on learning how to detect the status of the credit card in the normal cases (without fraud) due to the presence of a large number of normal transactions in the used dataset. Nevertheless, there are some factors that help finding the best ANN model for detecting fraud or normal transactions as the results show that ANN with two hidden layers, especially 29-15-8-2 model, has slightly higher accuracy from the other models.

4.5. Summary of the results

We started building the ANN models with one hidden layer and ended with 5 hidden layers. We can notice that the best model was obtained by using two hidden layers; the model is 29-15-8-2 and obtained

99.950% classification F-measure. Using one hidden layer achieves good results, but when another hidden layer has been added, the best classification F-measure has slightly risen from 84.757% to 85.129%. Adding three hidden layers lowered the F-measure and did not improve the results.

These results can be explained as: If you have too few hidden layers, you will get high training error and high generalization error due to under fitting. This part deals with the summary results collected during experiments. Table 7 compares results between the best ANN model in the one hidden layer, two hidden layers, and three hidden layers. Parameters chosen for comparison of results are F-measure, precision, sensitivity, specificity, accuracy, AUC, and RMSE. We notice that in all cases the results are remarkably close to each other, and this is expected due to the structure of the ANN. This is because that the ANN works on learning how to detect the status of the credit card in the normal cases (without fraud) due to the presence of a large number of normal transactions in the used dataset. Nevertheless, there are some factors that help finding the best ANN model for detecting fraud or normal transactions as the results show that ANN with two hidden layers, especially 29-15-8-2 model, has slightly higher accuracy from the other models.

Table 7. Summary of results regarding multiple hidden layers

Number of hidden Layers	1	2	3
Best Network Model	29-14-2	29-15-8-2	29-15-8-4-2
Accuracy	99.9505%	99.9515%	99.9414 %
Root mean square error (RMSE)	0.0218	0.0214	0.0234
Area under curve (AUC)	0.8983	0.9014	0.9003
F-measure	99.949%	99.950%	99.940%
Precision	99.949%	99.950%	99.940%
Sensitivity	99.950%	99.952%	99.941%
Specificity	79.710%	80.319%	80.116%

The F-measure comparison for each model is shown in Figure 2 which showed the high convergence of results between the three models in the different layers, but there was a very slight superiority for the model with two hidden layers for fraud detection. In all three models, ANN with two hidden layers performs better for fraud class detection, as shown in Figures 3-5. In comparison to one hidden layer and three hidden layers, it provides the highest F-measure in all three models. However, ANN with three hidden layers showed the lowest F-measure in all the three models as compared to the other two models in Figure 2. Figures 3-5 show the same results of ANN with two hidden layers for other performance measurement parameters (precision, sensitivity, and specificity).

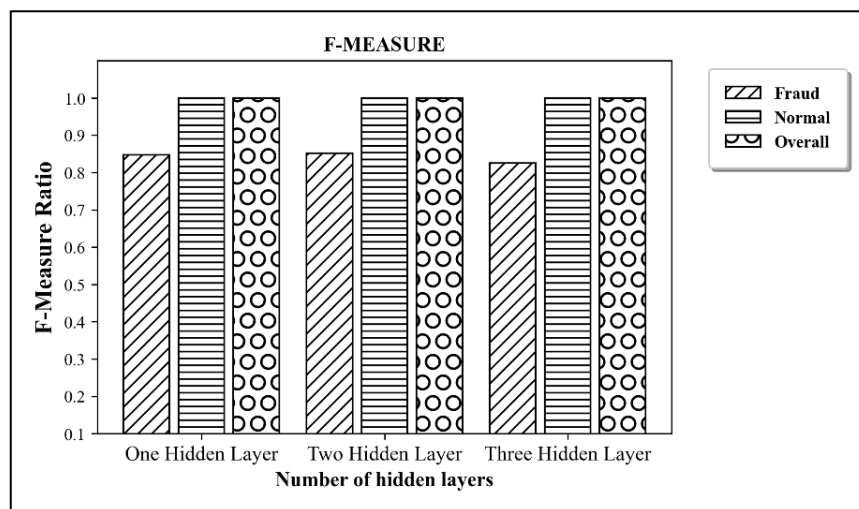


Figure 2. F-Measure

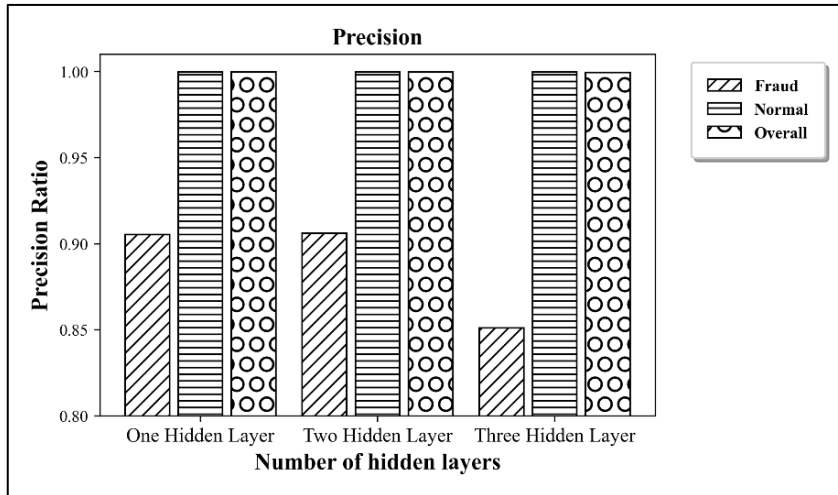


Figure 3. Precision

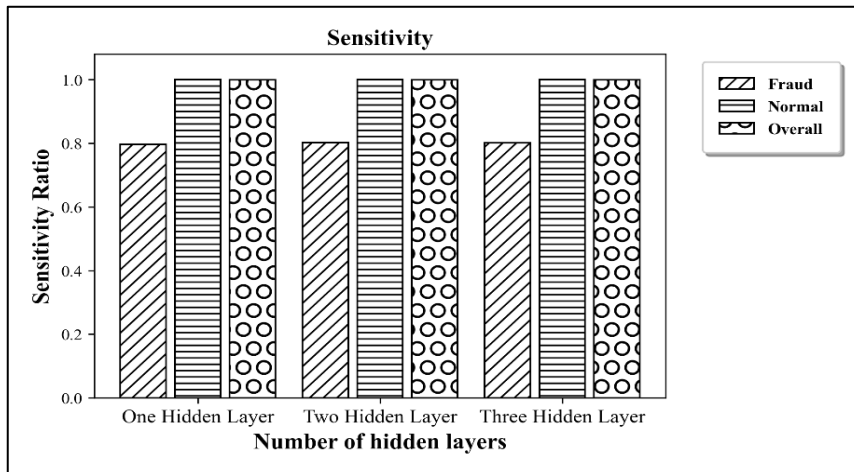


Figure 4. Sensitivity

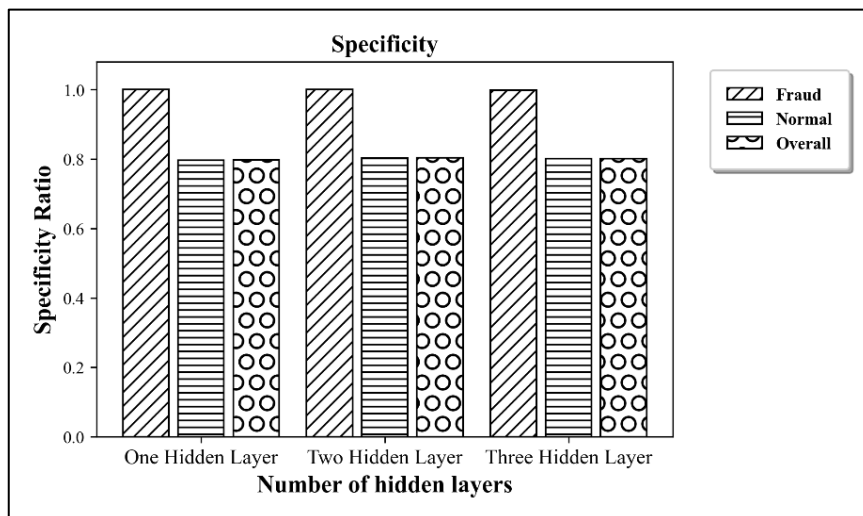


Figure 5. Specificity

The error rate of all methods is shown in Figure 6 using root mean square error (RMSE). This figure shows that ANN with two hidden layers was the best. According to the results of F-measure, Precision, Sensitivity, Specificity, and RMSE in Figures 2–6, the use of an ANN with two hidden layers model outperforms other ANN models in detecting fraud in credit card financial transactions.

Furthermore, Figure 7 reveals more about the best models chosen from each combination of hidden layers. The ROC figure that plots the relationship between the false positive rate and the true positive rate of the fraud class shows excellent results for the three models. As shown the value for all models starts from 0.8 at least, which gives an indication that any selected model will be able to detect fraud transactions with high accuracy.

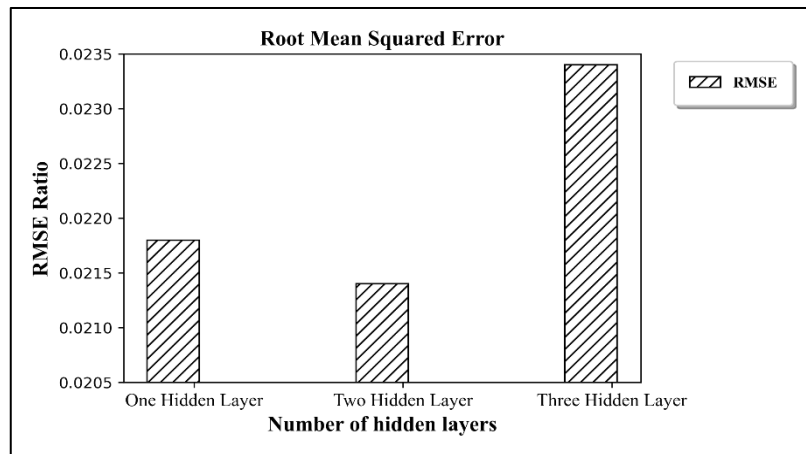


Figure 6. Root mean squared error

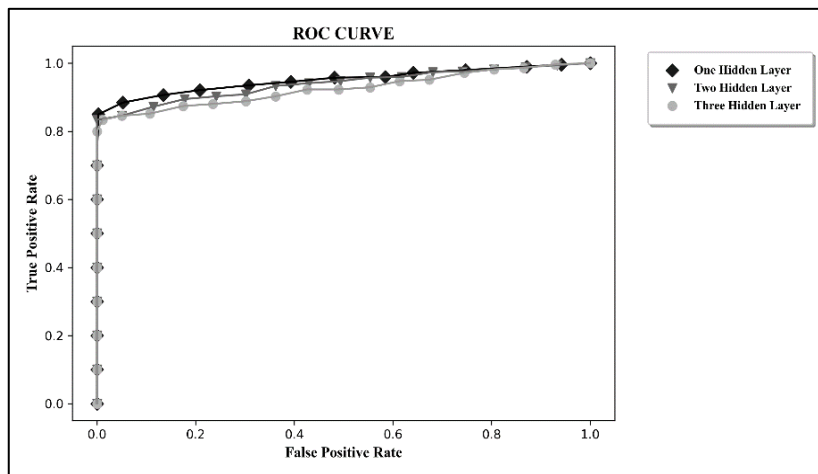


Figure 7. ROC curve

5. CONCLUSION

In this paper, we investigated the problem of fraud detection in credit card payment systems. We form the problem as a binary classification problem. To solve the problem, we choose to find the best model in ANN considering many variations in terms of hidden layers and the number of neurons in each one in an iterative way. After that, we conducted the experiments on a real dataset and discussed the results in term of F-measure. Results show that using ANN succeeded in getting high accuracy in the various layers used where the F-measure for ANN with one hidden layer the percentages of classification F-measure for Fraud cases were 84.76%, 85.13%, and 82.51% in one hidden layer, two hidden layers, and three hidden layers, respectively. From these results, it can be concluded that ANN is very useful in detecting fraud in credit card transactions. As future work, we will further investigate the problem of imbalanced credit card dataset.




REFERENCES

- [1] B. Lebichot, Y.-A. Le Borgne, L. He-Guelton, F. Oblé, and G. Bontempi, "Deep-learning domain adaptation techniques for credit cards fraud detection," in *INNS Big Data and Deep Learning conference*, 2019, pp. 78–88, doi: 10.1007/978-3-030-16841-4_8.
- [2] S. Kannan and K. Somasundaram, "Selection of optimal mining algorithm for outlier detection - an efficient method to predict/detect money laundering crime in finance industry," *Elysium Journal of Engineering Research and Management*, vol. 1, no. 1, pp. 30–42, 2014.
- [3] L. Seyedhossein and M. R. Hashemi, "Mining information from credit card time series for timelier fraud detection," in *2010 5th International Symposium on Telecommunications, IST 2010*, Dec. 2010, pp. 619–624, doi: 10.1109/ISTEL.2010.5734099.
- [4] E. W. T. Ngai, Y. Hu, Y. H. Wong, Y. Chen, and X. Sun, "The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature," *Decision Support Systems*, vol. 50, no. 3, pp. 559–569, Feb. 2011, doi: 10.1016/j.dss.2010.08.006.
- [5] A. Thennakoon, C. Bhagyani, S. Premadasa, S. Mihiranga, and N. Kuruwitaarachchi, "Real-time credit card fraud detection using machine learning," in *Proceedings of the 9th International Conference On Cloud Computing, Data Science and Engineering, Confluence 2019*, Jan. 2019, pp. 488–493, doi: 10.1109/CONFLUENCE.2019.8776942.
- [6] A. Dal Pozzolo, O. Caelen, Y. A. Le Borgne, S. Waterschoot, and G. Bontempi, "Learned lessons in credit card fraud detection from a practitioner perspective," *Expert Systems with Applications*, vol. 41, no. 10, pp. 4915–4928, Aug. 2014, doi: 10.1016/j.eswa.2014.02.026.
- [7] A. D. Pozzolo, G. Boracchi, O. Caelen, C. Alippi, and G. Bontempi, "Credit card fraud detection: A realistic modeling and a novel learning strategy," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 8, pp. 3784–3797, Aug. 2018, doi: 10.1109/TNNLS.2017.2736643.
- [8] Y. Lucas and J. Jurgovsky, "Credit card fraud detection using machine learning: A survey," *Proceedings of the International Conference on Intelligent Computing and Control Systems, ICICCS 2020*, Oct. 2020, pp. 1264–1270, doi: 10.1109/ICICCS48265.2020.9121114.
- [9] F. Carcillo, Y. A. Le Borgne, O. Caelen, Y. Kessaci, F. Oblé, and G. Bontempi, "Combining unsupervised and supervised learning in credit card fraud detection," *Information Sciences*, vol. 557, pp. 317–331, May 2021, doi: 10.1016/j.ins.2019.05.042.
- [10] U. Fiore, A. De Santis, F. Perla, P. Zanetti, and F. Palmieri, "Using generative adversarial networks for improving classification effectiveness in credit card fraud detection," *Information Sciences*, vol. 479, pp. 448–455, Apr. 2019, doi: 10.1016/j.ins.2017.12.030.
- [11] I. Sadgali, N. Sael, and F. Benabbou, "Bidirectional gated recurrent unit for improving classification in credit card fraud detection," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 21, no. 3, pp. 1704–1712, Mar. 2021, doi: 10.11591/ijeecs.v21.i3.pp1704-1712.
- [12] J. T. S. Quah and M. Sriganesh, "Real-time credit card fraud detection using computational intelligence," *Expert Systems with Applications*, vol. 35, no. 4, pp. 1721–1732, Nov. 2008, doi: 10.1016/j.eswa.2007.08.093.
- [13] E. Aleskerov, B. Freisleben, and B. Rao, "CARDWATCH: A neural network based database mining system for credit card fraud detection," in *IEEE/IAFE Conference on Computational Intelligence for Financial Engineering, Proceedings (CIFER)*, 1997, pp. 220–226, doi: 10.1109/cifer.1997.618940.
- [14] R. Brause, T. Langsdorf, and M. Hepp, "Neural data mining for credit card fraud detection," in *Proceedings of the International Conference on Tools with Artificial Intelligence*, Jul. 1999, pp. 103–106, doi: 10.1109/tai.1999.809773.
- [15] T. Guo and G. Y. Li, "Neural data mining for credit card fraud detection," in *Proceedings of the 7th International Conference on Machine Learning and Cybernetics, ICMLC*, Jul. 2008, vol. 7, pp. 3630–3634, doi: 10.1109/ICMLC.2008.4621035.
- [16] T. K. Behera and S. Panigrahi, "Credit card fraud detection: A hybrid approach using fuzzy clustering and neural network," in *Proceedings - 2015 2nd IEEE International Conference on Advances in Computing and Communication Engineering, ICACCE 2015*, May 2015, pp. 494–499, doi: 10.1109/ICACCE.2015.33.
- [17] C. Wang, Y. Wang, Z. Ye, L. Yan, W. Cai, and S. Pan, "Credit card fraud detection based on whale algorithm optimized BP neural network," in *13th International Conference on Computer Science and Education, ICCSE 2018*, Aug. 2018, pp. 614–617, doi: 10.1109/ICCSE.2018.8468855.
- [18] K. Fu, D. Cheng, Y. Tu, and L. Zhang, "Credit card fraud detection using convolutional neural networks," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9949 LNCS, 2016, pp. 483–490.
- [19] A. Roy, J. Sun, R. Mahoney, L. Alonzi, S. Adams, and P. Beling, "Deep learning detecting fraud in credit card transactions," in *2018 Systems and Information Engineering Design Symposium, SIEDS 2018*, Apr. 2018, pp. 129–134, doi: 10.1109/SIEDS.2018.8374722.
- [20] S. Georgieva, M. Markova, and V. Pavlov, "Using neural network for credit card fraud detection," in *AIP Conference Proceedings*, 2019, vol. 2159, p. 030013, doi: 10.1063/1.5127478.
- [21] S. Maes, K. Tuyls, B. Vanschoenwinkel, and B. Manderick, "Credit card fraud detection using bayesian and neural networks," *Maciunas RJ, editor. Interactive image-guided neurosurgery. American Association Neurological Surgeons*, pp. 261–270, 1993.
- [22] S. Panigrahi, A. Kundu, S. Sural, and A. K. Majumdar, "Credit card fraud detection: A fusion approach using Dempster-Shafer theory and Bayesian learning," *Information Fusion*, vol. 10, no. 4, pp. 354–363, Oct. 2009, doi: 10.1016/j.inffus.2008.04.001.
- [23] A. Srivastava, A. Kundu, S. Sural, and A. K. Majumdar, "Credit card fraud detection using Hidden Markov Model," *IEEE Transactions on Dependable and Secure Computing*, vol. 5, no. 1, pp. 37–48, Jan. 2008, doi: 10.1109/TDSC.2007.70228.
- [24] S. Bhattacharyya, S. Jha, K. Tharakunnel, and J. C. Westland, "Data mining for credit card fraud: A comparative study," *Decision Support Systems*, vol. 50, no. 3, pp. 602–613, Feb. 2011, doi: 10.1016/j.dss.2010.08.008.
- [25] Y. Sahin and E. Duman, "Detecting credit card fraud by decision trees and support vector machines," *IMECS 2011 - International MultiConference of Engineers and Computer Scientists 2011*, vol. 1, pp. 442–447, 2011.
- [26] A. C. Bahnsen, D. Aouada, and B. Ottersten, "Example-dependent cost-sensitive decision trees," *Expert Systems with Applications*, vol. 42, no. 19, pp. 6609–6619, Nov. 2015, doi: 10.1016/j.eswa.2015.04.042.
- [27] E. Duman and M. H. Ozelik, "Detecting credit card fraud by genetic algorithm and scatter search," *Expert Systems with Applications*, vol. 38, no. 10, pp. 13057–13063, Sep. 2011, doi: 10.1016/j.eswa.2011.04.110.
- [28] S. Jha, M. Guillen, and J. Christopher Westland, "Employing transaction aggregation strategy to detect credit card fraud," *Expert Systems with Applications*, vol. 39, no. 16, pp. 12650–12657, Nov. 2012, doi: 10.1016/j.eswa.2012.05.018.
- [29] D. Sánchez, M. A. Vila, L. Cerda, and J. M. Serrano, "Association rules applied to credit card fraud detection," *Expert Systems with Applications*, vol. 36, no. 2 PART 2, pp. 3630–3640, Mar. 2009, doi: 10.1016/j.eswa.2008.02.001.




- [30] A. D. Pozzolo, O. Caelen, R. A. Johnson, and G. Bontempi, "Calibrating probability with undersampling for unbalanced classification," in *Proceedings-2015 IEEE Symposium Series on Computational Intelligence, SSCI 2015*, Dec. 2015, pp. 159–166, doi: 10.1109/SSCI.2015.33.
- [31] F. Itoo, Meenakshi, and S. Singh, "Comparison and analysis of logistic regression, Naïve Bayes and KNN machine learning algorithms for credit card fraud detection," *International Journal of Information Technology (Singapore)*, vol. 13, no. 4, pp. 1503–1511, Aug. 2021, doi: 10.1007/s41870-020-00430-y.
- [32] M. Alkhalili, M. H. Qutqut, and F. Almasalha, "Investigation of applying machine learning for watch-list filtering in anti-money laundering," *IEEE Access*, vol. 9, pp. 18481–18496, 2021, doi: 10.1109/ACCESS.2021.3052313.
- [33] A. Özdemir, U. Yavuz, and F. A. Dael, "Performance evaluation of different classification techniques using different datasets," *International Journal of Electrical and Computer Engineering*, vol. 9, no. 5, pp. 3584–3590, Oct. 2019, doi: 10.11591/ijece.v9i5.pp3584-3590.
- [34] A. Krenker, J. Bester, and A. Kos, "Introduction to the artificial neural networks," *Artificial Neural Networks - Methodological Advances and Biomedical Applications*, pp. 1–18, 2011, doi: 10.5772/15751.
- [35] H. Ramchoun, M. Amine, J. Idrissi, Y. Ghanou, and M. Ettaouil, "Multilayer perceptron: architecture optimization and training," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 4, no. 1, p. 26, 2016, doi: 10.9781/ijimai.2016.415.
- [36] S. Xu and L. Chen, "A novel approach for determining the optimal number of hidden layer neurons for FNN's and its application in data mining," *5th International Conference on Information Technology and Applications, ICITA 2008*, 2008, pp. 683–686.
- [37] G. Wu and E. Y. Chang, "Class-boundary alignment for imbalanced dataset learning," *ICML Workshop on Learning from Imbalanced Data Sets II*, pp. 49–56, 2003.
- [38] B. L. Sturm, "Classification accuracy is not enough: On the evaluation of music genre recognition systems," *Journal of Intelligent Information Systems*, vol. 41, no. 3, pp. 371–406, Dec. 2013, doi: 10.1007/s10844-013-0250-y.
- [39] S. Boughorbel, F. Jarray, and M. El-Anbari, "Optimal classifier for imbalanced data using Matthews correlation coefficient metric," *PLoS ONE*, vol. 12, no. 6, p. e0177678, Jun. 2017, doi: 10.1371/journal.pone.0177678.

BIOGRAPHIES OF AUTHORS






Bassam Kasasbeh    received his master's degree from University of Jordan in Computer Science. He is currently a lecturer of at Computer Science department at Applied Science Private University. His research interests include machine learning, Cyber security, Internet of things. He can be contacted at email: b_kasasbeh@asu.edu.jo.



Balqeas AL-Dabaybah    CS Master degree graduated in 2018 from Princess Sumaya University for technology (PSUT). My research interest in machine learning, cloud computing, algorithms and HCI. She can be contacted at email: b_aldabaybah@asu.edu.jo.



Hadeel Ahmad    received the B.Sc. degree in computer Science from Applied Science University, Jordan, in 2001, the M.Sc. degree in computer systems from Arab Academy for banking and financial science, Jordan, in 2004. She is currently a lecturer with the Faculty of Information Technology, Applied Science University, Jordan, since October 2018. Her research interests are in machine learning, semantic web, text mining and ontologies. She can be contacted at email: h_ahmad@asu.edu.jo.