Detecting Credit Card Fraud using Machine Learning

https://doi.org/10.3991/ijim.v15i24.27355

Arjwan H. Almuteer[1], Asma A. Aloufi[1], Wurud O. Alrashidi[1], Jowharah F. Alshobaili[1], Dina M. Ibrahim[1,2(✉)]
[1]Department of Information Technology, College of Computer, Qassim University, Buraydah, Saudi Arabia
[2]Computers and Control Engineering Department, Faculty of Engineering, Tanta University, Tanta, Egypt
dina.mahmoud@f-eng.tanta.edu.eg

Abstract—Credit card is getting increasingly more famous in budgetary exchanges, simultaneously frauds are likewise expanding. In the past, fraud practitioners were identified using rule-based master frameworks, which ignored a variety of variables, including the outlandishly imbalanced nature of positive and negative cases. Using named information, we provide an approach to fraud detection that uses Convolutional Neural Networks (CNNs) and is based on CNNs. An element lattice speaks to a plethora of interchange information and uses a convolutional neural organization to recognize a large number of idle examples for each of those examples. A considerable business bank's boss presentation is compared with several best-in-class techniques in trials on truly monstrous exchanges. Our objective is to combine CNN with LSTM and Auto-encoder to increase credit card fraud detection while improving the previous models' performance. By using these four models; CNN, AE, LSTM, and AE&LSTM. each of these models is trained by different parameter values highest accuracy has been achieved where the AE model has accuracy = 0.99, the CNN model has accuracy = 0.85, the accuracy of the LSTM model is 0.85, and finally, the AE&LSTM model obtained an accuracy of 0.32 by 400 epoch. It is concluded that the AE classifies the best result between these models.

Keywords—fraud detection, CNN, LSTM, auto encoder

1 Introduction

The detection of credit card fraud has recently spread due to increased fraud that can be described as a deliberate ploy committed to achieve some kind of gain, usually based on money. It's an unfair practice that's becoming increasingly happening day after day. In recent times the use of electronic devices in payment methods such as credit cards, as a result of which, credit card fraud has increased [1], as the majority of people have become shopping through the Internet and pay and pay their bills from credit cards and pay as well, and people can get money and transfer them using their online banking systems. All this technology our lives are easier and faster, despite all these

positive aspects. This technology has brought a significant risk in terms of unauthorized payments, known as financial frauds. We can express these banking transactions as fraud and online identity theft, and fraud on payment cards may amount to money laundering [2]. Prior to the widespread adoption of machine learning and deep learning, it was difficult for banks and businesses to classify fraud in detection systems. The problem is addressed by converting the fraud problem into a binary classification file, thanks to advances in learning supervision. Suitable for high-risk transactions. Many researchers have gathered financial data through networks and banks in recent years in order to build a fraud detection system. These algorithms' goal is to build a supervised learning model in the data set without being explicitly programmed [3].

Credit card fraud is a modern problem of our time as technology develops. In this chapter, we talk about the most important scientific terms related to fraud. We also study and analyze the history of experiments and research of scholars and experts on fraud, after which the proposed system will be presented and discussed briefly Credit card fraud can be described as illegal usage of Mastercard information for online purchase. Charge card trades are done truly or basically. Physical trades imply trades incorporating physical collaboration with shipper. Customers are expected to present a physical card at the reason for acquirement. Virtual trades suggest trades performed over the web or telephone. It anticipates that customers should give certain card information, (for instance, CVV number, mystery key, security question, etc.) for online purchases. The advancement of Visas has not recently made online trades reliable, more straightforward, pleasing and beneficial, it has moreover given new deception events to criminals, and extended the movement of blackmail. The effect of Mastercard distortion is upsetting and it is having affected the overall economy in quantifiable habits. An enormous number of US dollar has been lost by various individuals and organizations. In 2009, the full-scale assessment of online solicitation (for product and ventures just) was generally US$15 billion. Also, 84% of these solicitations were paid on the web. In 2013, coercion was surveyed to cost US retailers about $23 billion, and in 2014, the cost of deception rose to generally $32 billion [4].

2 Background

There are many benefits to using a credit card in everyday life, but there are also many drawbacks, the most significant of which is the risk of fraud. To mitigate this risk, some machine learning methods can be employed to collecting data, which is what we really discuss in this article [1].

2.1 Convolutional neural network (CNN)

Initially we give a portray for CNN-based fraud detection framework as illustrated in Figure 1.

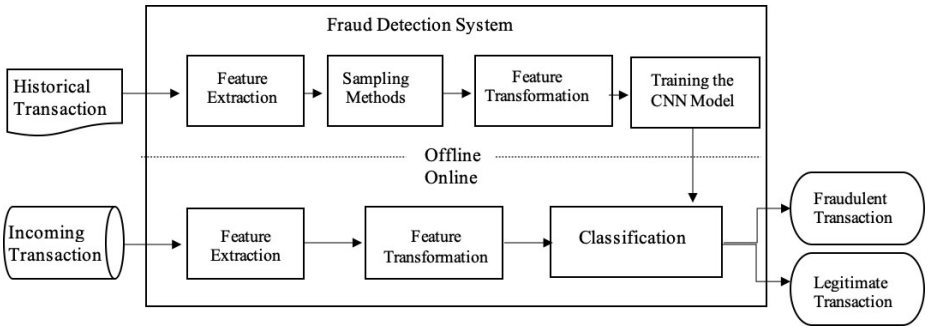

Fig. 1. The representation of credit card fraud detection system

The CNN model is suitable for preparing a large amount of data, and it has the tool to avoid the model over-fit. Image order and discourse signal processing are two examples of disciplines where convolutional neural networks have found success. There is a total of six different levels in this system. The input is a feature matrix. Additionally, the first layer is a convolutional one, followed by a sub testing one. On top of it, there's a convolutional layer for good measure. As an added bonus, the final three layers are all located on the complete association layer. Figure 2 shows the structure of the CNN model [5].

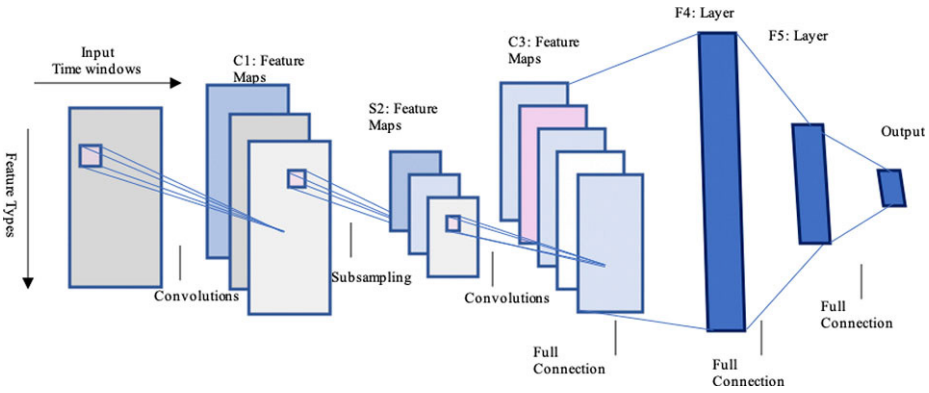

Fig. 2. The structure of CNN model

Compared to conventional CNN the precision rate of traditional BP neural networks with two hidden layers can be stabilized at around 91%. When we talk about sequencing, we're referring to the complete network. Layers for pooling and a fully connected layer with four consecutive convolutional distribution and distributed layers the convolutional layer method extracts and automatically extracts the input data's local features. Derivative qualities are new features associated with the input characteristics, but their physical characteristics are not explained. Context does help the model classify things. The layer of pooling combines the properties of nearby regions. They work together to reduce information redundancy by performing a single higher-level function. In the final classification's position, the layer that is fully connected plays a role. There is a 65

second training process for the element sequencing convolutional neural network, as opposed to the other two convolutional neural networks. It takes 752 seconds to train a model with ten different feature combinations. It takes 352 seconds to train a CNN without using the feature derivative.

The model's overall performance is even worse than their method whenever we have included the computation of the generated features. The current CNN model is linked to the CNN based on feature sequencing. A number of experiments have shown that the new model presented in this study outperforms the conventional CNN model depending on the outcome of each forecast and does not require a huge number of derivatives. It does not require high-dimensional input features or derivative variables and can obtain a generally well-ordered input arrangement in a small number of iterations [6].

2.2 Long short-term memory (LSTM)

LSTM (Long Short-Term Memory) was developed in 1997. LSTM is much of the time utilized as a model to tackle issues in AI 28. It is intended to deal with long haul reliance issue that vanilla Recurrent Neural Network (RNN) can't oblige 29. Our inspiration in utilizing LSTM is to separate the data from successive information [2].

2.3 Auto-encoder (AE)

Autoencoder (AE) is a form of ANN (artificial neural network) is an algorithm that spread over backpropagation by placing inputs equivalent with outputs. As in Figure 3, it contains a hidden inner layer that defines the code used to describe the entry, and consists of two main elements: the first is encoding: it identifies the value of the entry in the code, and the other decoding: it draws a code to rebuild the primary portal as in. Autoencoder (AE) has several different types depending on their inclusion of hidden interior layers, The most important Encoded Automatic Noise Reduction. As illustrated in Figure 4.

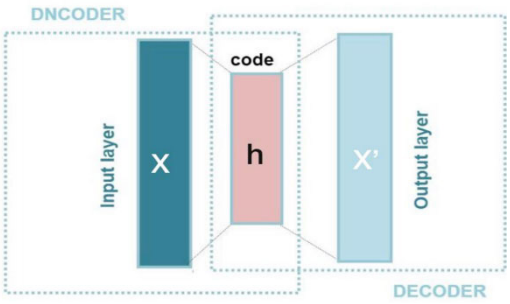

Fig. 3. The schema of a basic auto-encoder

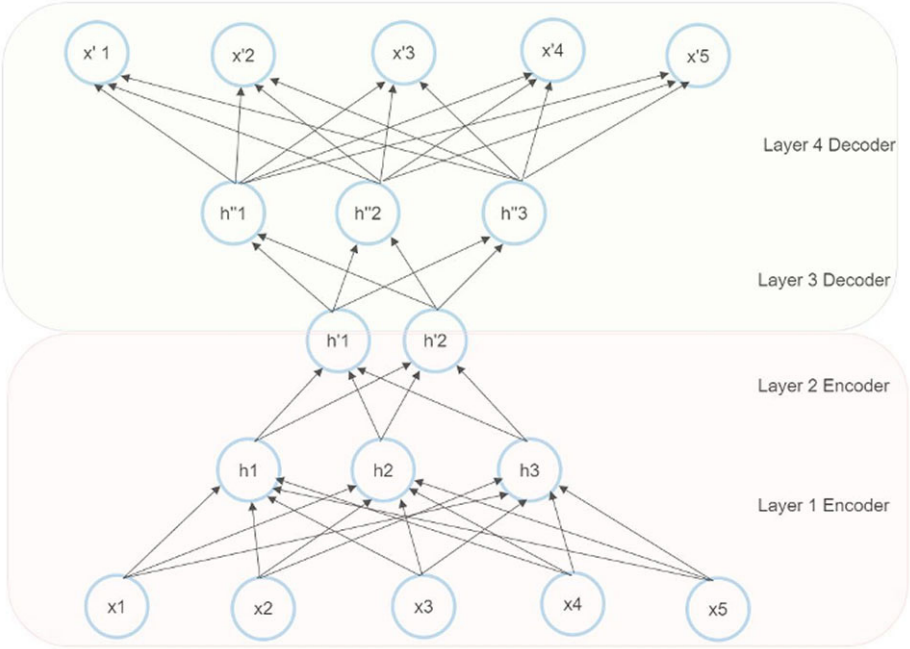

Fig. 4. The auto-encoder with 4 layers

3 Literature review

During our research, we read many scientific papers that specialized in detecting fraud on the credit card, and We collected the most important information in Table 1. Now we will talk about some of the research that will help us with our work.

Table 1. Previous Study of credit card fraud detection methods

Ref	Method	Dataset	Accuracy	Sensitivity	Precision
[7]	Logistic regression, Decision tree SVM, Random Forest	284,786 transactions from Group of ULB (Universite Libre de Bruxelles).	– Random Forest = 0.986 – Logistic reg.= 97.7 – SVM = 97.5 – Decision tree = 95.5	–	–
[3]	NN, SVM, Naive Bayes, Logistic Regression	CIS based IEEE dataset that available on Kaggle.	– NN = 0.954 – NN with focal loss = 0.957 – SVM = 0.932 – Naïve Bayes = 0.875 – Logistic Regression = 0.911	–	–

(Continued)

Table 1. Previous Study of credit card fraud detection methods *(Continued)*

Ref	Method	Dataset	Accuracy	Sensitivity	Precision
[8]	Auto Encoder, RBM,	use three datasets; German, Australian, and European datasets downloaded on Kaggle platform	– AE = 0.9603 – RBM = 0.9505	–	–
[9]	CNN SVM LR CNN MLP DEAL	284,807 transactions 492 fraudulent transactions 284,315 legitimate transactions	– SVM = 67 – LR= 63 – CNN = 82 – MLP = 81 – AE = 83.67 – DEAL = 99.99.8	–	SVM = 99.94 LR = 99.91 CNN = 99.89 MLP :99.94 AE = 96.03 DEAL = 99.81
[1]	KNN, DT, NB, RF Logistic Regression	Dataset of European cardholders From Group of ULB (Universite Libre de Bruxelles)	–	– Decision Tree = 79.21 – KNN = 91.11 – Logistic Reg. = 87.67 – Random forest = 93.83 – Naive Bayes = 6.56	– Decision Tree = 85.11 – KNN = 81.19 – LR. = 65.34 – RF = 75.25 – Naive Bayes = 85.15
[10]	Light GBM algorithm	The first dataset: 284,807 credit card transactions Second dataset:94,683 transactions.	– 1st dataset = 90.94 – 2nd dataset = 92.90	– Dataset1 = 97.34 – Dataset2 = 91.72%	–
[11]	SVM KNN	462279 are non-fraud transactions and 5417 fraudulent transactions	– SVN = 85.5% – KNN = 82.0%	– SVM = 0.908	– SVM = 97.6 – KNN = 81.2%
[12]	Random forest Isolation forest, Logistic regression, Decision tree	284807 transactions, where 492 of them are frauds.	RF = 0.99% IF = 0.58% LR = 0.97% Decision tree = 0.97%	–	RF = 0.9310 IF = 0.0147 LR = 0.875 Decision tree = 0.8854
[13]	MLP ELM	284807 transactions, where 492 of them are frauds.	MLP = 97.84 ELM = 95.46	–	MILP = 99.32 ELM = 98.83
[14]	Naive Bayes KNN Logistic Regression Multilayer Perceptron Ada Boost	284807 transactions, where 492 of them are frauds.	Logistic Regression = 98.2% Naive Bayes = 99.6% K-NN = 94.4% MLP = 98.4% Ada Boost = 98.5%	–	Logistic Reg. = 0.07 Naive Bayes = 0.26 KNN = 0.02 Multilayer Perceptron = 0.08 Ada Boost = 0.09

Pipe-ling and ensemble learning were proposed by the authors of [15] as a way to detect credit card fraud. Nine strategies were evaluated and compared for this work, including RF, LR, NB and KNN, as well as MLP and pipelining. Nine Method was applied to a database containing 284,807 anonymized transactions to extract the best of method in fraud detection depending on time and quantity, and compared to Accuracy where the nine Methodists were as follows: KNN 94.4%, Quadrant Discriminant Analysis 97.3%, Multilayer Perception 98.4%, Logistic 98.2% , Naive Bayes 99.6%, Ada Boost 98.5%, Random Forest 99.7%, Pipelining 99.99%, Ensemble Learning 99.99%, Which led to Pipelining outperforming the rest.

Similarly, the authors of [7] present a detection method for credit card fraud constructed by machine learning models and the collection of machine learning models. While researching several credit card fraud detection systems, the authors of this work compiled data to find the most effective algorithm for solving their problem. We used each method of LR, DT, SVM and RF on database consisting of 284,786 transactions to be the result of my random forest achieved the best value as the of Precision = 0.997, Sensitivity = 0.984 and the Accuracy = 0.986.

To improve performance, the paper proposes a deep neural network-based algorithm for credit card detection in [3]. We employ four methods: NN, SVM, Nave Bayes, and Logistic Regression. On both scores, the neural network model (NN) outperforms the other models. NN has a precision of 0.954. NN accuracy with (focal loss) = 0.957. The NN with a focal loss has a higher fraud detection accuracy, implying that the focal loss can improve NN model training. The work on [9] suggest data imbalance, they proposed framework is presented as being two main stages are involved: learning and prediction. The learning is offline and includes transformation of functions, normalization, training of models, and optimization. To build the model of Deep Ensemble Learning, To fit into the model, the characteristics are converted to tensors and they applied Extra-Tree Ensemble in and class derived using DNN are improved through the ensemble method in this path, but in the prediction phase the online transaction characteristics are transformed into a tensor and Supplied to the model proposed. To mark the transaction as fraudulent or legitimate, the model generates the alarm, the accuracy of their model up 99.99.8% and proposed comparison of the proposed DEAL with benchmark ML classifiers, LR and SVM, and DL models, CNN, MLP, and, Auto-Encoder (AE), SVM 67%, LR 63% ,CNN 82%, MLP 81%, AE 67%, and DEAL 99.8%.

In the study of [12], Using sliding window methodology, the authors proposed and developed a new fraud detection approach for streaming transactional data with the goal of extracting behavioral patterns from prior customer transaction details. After that they applied different Methods the Random forest is higher accuracy up to 0.99% and others Local Outlier factor and Accuracy up to 45% and Isolation forest is 0.58%, Logistic regression is 0.97%, Decision tree 0.97%. Monitoring for Credit Card Theft They used two different classification techniques from artificial neural networks focused on the Multilayer Perceptron and Extreme Learning Machine Structures to detect fake transactions on the credit card fraud dataset with an accuracy of up to 97.84 percent: the Multilayer Perceptron (MLP) and the Extreme Learning Machine (ELM). One hidden layer feedforward artificial neural network (SLFN) is used in Extreme Learning Machine, and the weights connecting the hidden nodes to their outputs are computed analytically. Only one neural network feeds information into it. MLP is a feed-forward

net that also uses the gradient descent Method to reduce the amount of the error evaluated by the mean absolute percentage error.

4 Experiments

This part presents the proposed fraud detection model and depict its stages, as in Figure 5. The model is divided into 3 phases to be specific input stage, during applying the model stage and output stage.

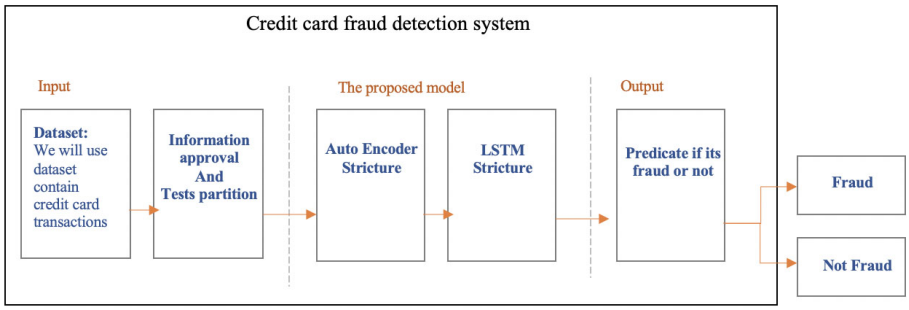

Fig. 5. The proposed credit card fraud detection

First the credit card dataset is taken details and the process of training from the selection of the coding environment and the preprocessing used on the dataset before balance and after. We explain two techniques that are used to deal with imbalance dataset. We divided into two parts data preprocessing and code implementation. In the Implementation of our project, we used the following tools: the first tool is the Colab which is a creation from google research it is a Jupyter notebook hosted service that provides free access to computing resources including GPUs (Graphics Processing Unit) and TPU (Tensor Processing Unit) industrialized by Google specifically for neural network machine learning. TPU considered to quicken deep learning methods. The second tool is Python has become popular programming language. we used to develop code to predict frauds using machine learning. Data Preprocessing The project aims to increase the accuracy and the imbalanced data should be solved before the training. In most cases, imbalance data refers to a classification problem in which the classes are not defined similarly. As shown in Figure 6, there is a very high difference between the number of Normal and fraud from our dataset which mean imbalanced data and that will affect the test result.

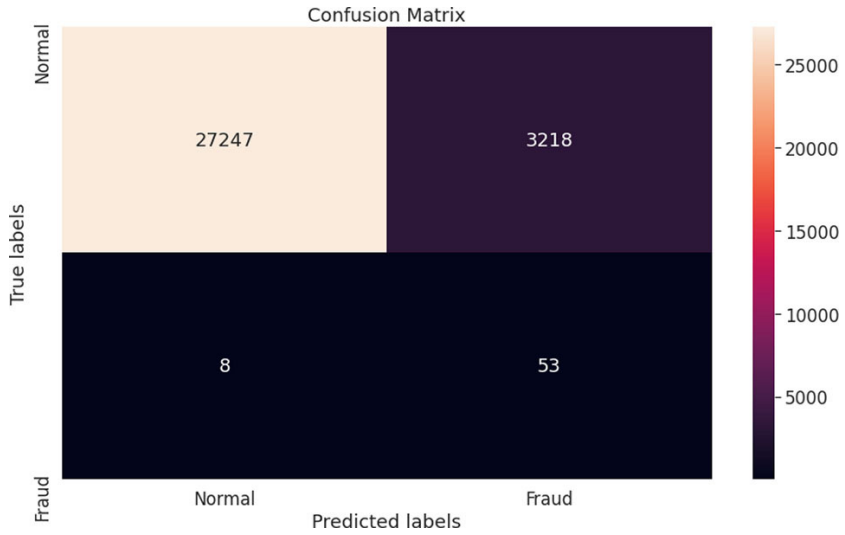

Fig. 6. The confusion matrices for imbalanced dataset

Imbalanced data affect the prediction by predict Credit card Fraud and Normal operation. There are two ways to deal with imbalanced data. Remove sample data from the major class due to Under-Sampling. The Over-Sampling, which duplicates samples from the minority, is also a concern. Figure 7 depicts the outcome after data balancing was applied.

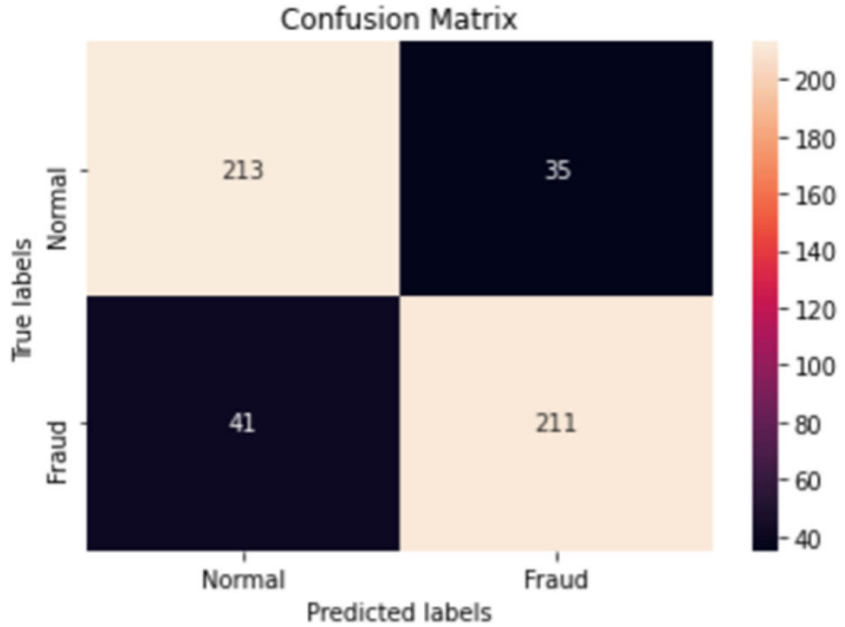

Fig. 7. The confusion matrices for balanced dataset

Through our experimentation with both under-Sampling and over-Sampling, it became clear to us that better and higher values using over-Sampling due to taking a large data sample, training it and testing it, which improves the accuracy ratio.

We employ Accuracy, precision, recall, F1-score, ROC, AUC, and a confusion matrix to assess the classification model's performance.

Accuracy: addressing the quantity of exact expectations for all forecasts, is additionally the establishment boundary utilized for model assessment.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

Precision: is the part of effectively predicted positive perceptions to the absolute expected positive perceptions.

$$precision = \frac{TP}{TP + FP} \tag{2}$$

Recall: is an estimation of the quantity of positive cases altogether the positive information which the classifier effectively anticipated.

$$recall = \frac{TP}{TP + FN} \tag{3}$$

F1-score: is the Precision and Recall weighted normal. This score subsequently considers both bogus positive and bogus negatives. It isn't as clear to get a handle on instinctively as accuracy, yet F1 is by and large more valuable than exactness, specifically if the class appropriation is lopsided. It shows the presentation of an arrangement model at all grouping limitations via a curve called the receiver operating characteristic curve (ROC curve). True Positive Rate and False Positive Rate are two of the parameters included.

$$F1 = 2.\frac{precision.recall}{precision + recall} \tag{4}$$

5 Results and discussion

This study was implemented in Google Colab Notebook, using python. It was not a successful experience and the Classify the AE best result with Over-Sampling technique, The our dataset is contain 284315 Normal transaction and 492 fraud transaction [16] [17] as we mentioned previously we used Under-Sampling and Over-Sampling to balanced dataset we use this technique with AE, CNN, LSTM, AE&LSTM and with different epoch value on Colab Notebook, the result after sampling dataset with Under-Sampling and Over-Sampling at epoch 400 shown in Table 2.

Table 2. Result of training at epoch 400

	Accuracy	Precision	Recall	F1 score	ROC AUC
After Under Sampling					
CNN	0.854000	0.856574	0.853175	0.854871	0.925211
AE	0.957006	0.036292	0.938776	0.069882	0.947907
LSTM	0.846000	0.848606	0.845238	0.846918	0.923851
AE&LSTM	0.282083	0.001710	0.752688	0.003412	0.532543
After Over Sampling					
CNN	0.846000	0.840467	0.857143	0.848723	0.925131
AE	0.999087	0.895522	0.571429	0.697674	0.785653
LSTM	0.848000	0.857724	0.837302	0.847390	0.923275
AE&LSTM	0.328124	0.011877	0.889045	0.003345	0.592543

As shown in Figure 8, the train and validation accuracy. The train accuracy and the validation accuracy significantly increased from 0 to 40 epoch and from 40 to 400 the train accuracy increased in Converging values but the validation accuracy is not fixed in the same value. The figure shows also the train and validation loss. The train loss decreased when the epoch value increase. The validation loss is not fixed. It is increased and decreased in different values. Figure 9 illustrates the train and validation accuracy. The train accuracy and the validation accuracy significantly increased from 0 to 50 epoch and from 50 to 400 the train accuracy increased in Converging values but the loss value is not fixed. It is increased and decreased in different values. The train loss decreased when the epoch value increase. The loss value decreased when the number of epochs value increase.

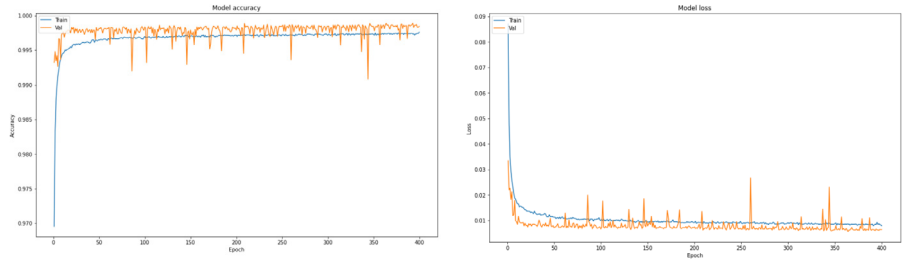

Fig. 8. The accuracy and loss for CNN training model

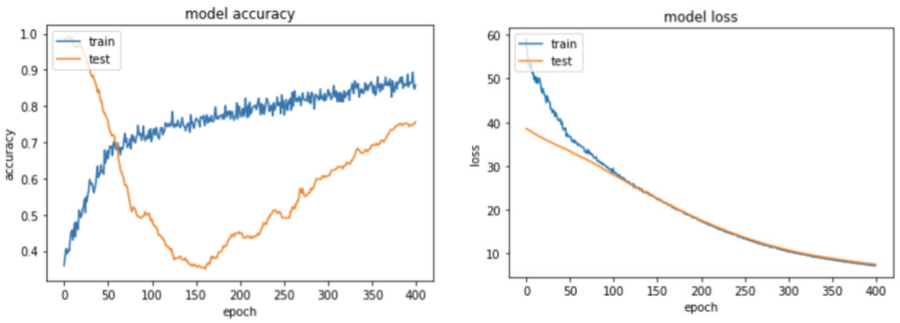

Fig. 9. The accuracy and loss for LSTM training model

Figure 10 illustrates the train and validation accuracy. The train accuracy and the validation accuracy significantly increased from 0 to 40 epoch and from 40 to 400 the train accuracy increased in Converging values but the validation accuracy is not fixed in the same value. It also shows the train and validation loss. The train loss decreased when the epoch value increases the loss value is not fixed. It is increased and decreased in different values. While in Figure 11 the train and validation accuracy. The train accuracy fixed from 0 to 50 but from 50 to 400 is not fixed some time decreased or increased. the accuracy value is not fixed in the same value. In addition, it shows the train and validation loss. The train loss fixed from 0 to 200 but from 200 to 400 is not fixed. The validation loss is not fixed. It is increased and decreased in different values. The difference between the three previous methods and our proposed is demonstrated in Figure 12. The four models: CNN, AE, LSTM, and AE&LSTM, have been trained by different parameter values and the highest test accuracy achieved AE = 0.99, CNN = 0.85, LSTM = 0.85 and AE&LSTM = 0.32 by 400 epochs. we conclude the AE classify best result between this models and AE&LSTM did not achieve good results.

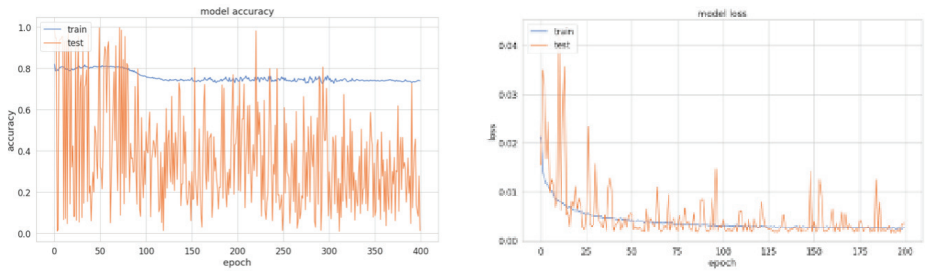

Fig. 10. The accuracy and loss for AE training model

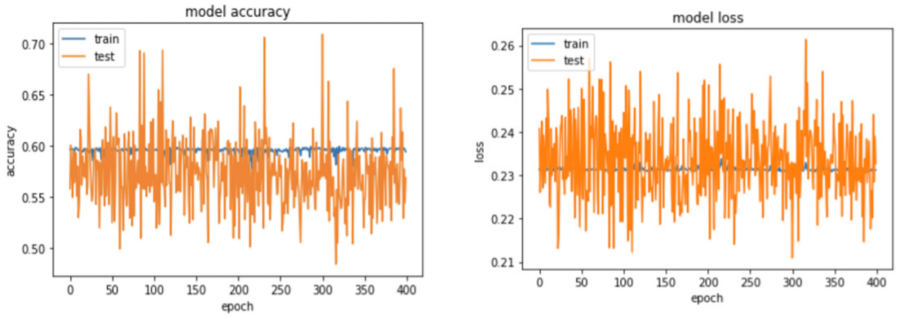

Fig. 11. The accuracy and loss for AE & LSTM training model

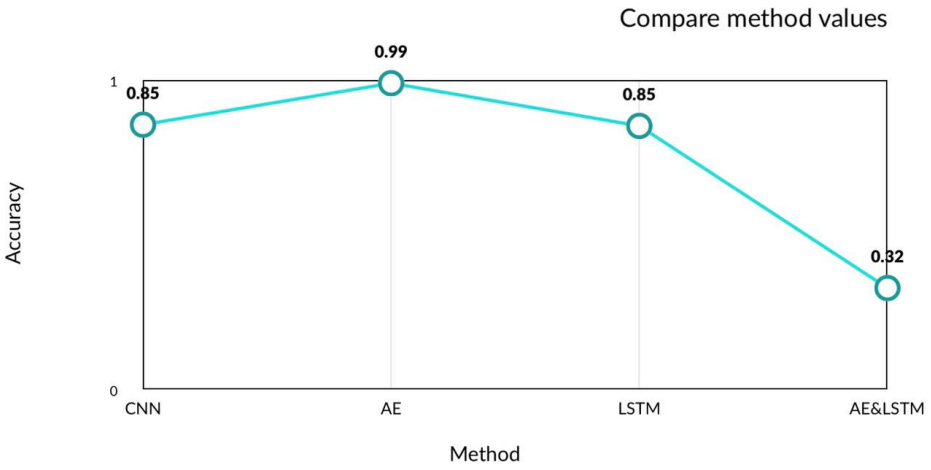

Fig. 12. The difference between the three previous methods and our proposed

6 Conclusion

The goal of this research is to combine between CNN, LSTM, and Auto-encoders (AEs) in order to progress credit card fraud detection and the performance of prior models. CNN, AE, LSTM, and AE&LSTM are four models that can be used. Different parameter values are used to train each of these models. The AE model has the highest accuracy, with an accuracy of 0.99, the CNN model has an accuracy of 0.85, the LSTM model has an accuracy of 0.85, and the AE&LSTM model has an accuracy of 0.32 after 400 epochs. The AE classifies the best outcome among these models, it is concluded. In the near future, we will explore some more technique [18–27] of security to detect and classify the fraud detection in credit card.

7 References

[1] Hussein, A. S., Khairy, R. S., Najeeb, S. M. M., and ALRikabi, H. T. (2021). Credit Card Fraud Detection Using Fuzzy Rough Nearest Neighbor and Sequential Minimal Optimization with Logistic Regression. International Journal of Interactive Mobile Technologies, 15(5). https://doi.org/10.3991/ijim.v15i05.17173

[2] Alghofaili, Y., Albattah, A., and Rassam, M. A. (2020). A Financial Fraud Detection Model Based on LSTM Deep Learning Technique. Applied Security Research, 15: 498–516. https://doi.org/10.1080/19361610.2020.1815491

[3] Yu, X., Li, X., Dong, Y., and Zheng, R. A Deep Neural Network Algorithm for Detecting Credit Card Fraud (2020). International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), June 12 2020, IEEE, pp. 181–183. https://doi.org/10.1109/ICBAIE49996.2020.00045

[4] Sadgali, I., Sael, N., and Benabbou, F. (2020). Adaptive model for credit card fraud detection. International Journal of Interactive Mobile Technologies, 14(3). https://doi.org/10.3991/ijim.v14i03.11763

[5] Fu, K., Cheng, D., Tu, Y., and Zhang, L. 2016. Credit card fraud detection using convolutional neural networks. In International conference on neural information processing, October 16 2016, Springer, Cham. pp. 483–490. https://doi.org/10.1007/978-3-319-46675-0_53
[6] Chen, K., Yadav, A., Khan, A., Meng, Y., and Zhu, K. (2019). Improved crack detection and recognition based on convolutional neural network. Modelling and simulation in engineering, 2019. https://doi.org/10.1155/2019/8796743
[7] Dornadula, V. N., and Geetha, S. (2019). Credit card fraud detection using machine learning algorithms. Procedia computer science, 165: 631–641. https://doi.org/10.1016/j.procs.2020.01.057
[8] Pumsirirat, A., and Yan, L. (2018). Credit card fraud detection using deep learning based on auto-encoder and restricted boltzmann machine. International Journal of advanced computer science and applications, 9: 18–25. https://doi.org/10.14569/IJACSA.2018.090103
[9] Arya, M., and Sastry G. H. (2020). DEAL–'Deep Ensemble ALgorithm' framework for credit card fraud detection in real-time data stream with Google TensorFlow. Smart Science, 8: 71–83. https://doi.org/10.1080/23080477.2020.1783491
[10] Taha, A. A., and Malebary, S. J. (2020). An intelligent approach to credit card fraud detection using an optimized light gradient boosting machine. IEEE Access, 8: 25579–25587. https://doi.org/10.1109/ACCESS.2020.2971354
[11] Singh, A., and Jain, A. (2019). Financial fraud detection using bio-inspired key optimization and machine learning technique. International Journal of Security and Its Applications, 13: 75–90. https://doi.org/10.33832/ijsia.2019.13.4.08
[12] Dornadula, V. N., and Geetha, S. (2019). Credit card fraud detection using machine learning algorithms. Procedia computer science, 165: 631–641. https://doi.org/10.1016/j.procs.2020.01.057
[13] El Hlouli, F. Z., Riffi, J., Mahraz, M. A., El Yahyaouy, A., and Tairi, H. Detection of SMS Spam Using Machine-Learning Algorithms. Embedded Systems and Artificial Intelligence (ESAI), Apr. 7 2020, Fez, Morocco, p. 429. https://doi.org/10.1007/978-981-15-0947-6_41
[14] Tingfei, H., Guangquan, C., and Kuihua, H. (2020). Using variational auto encoding in credit card fraud detection. IEEE Access, 8: 149841–149853. https://doi.org/10.1109/ACCESS.2020.3015600
[15] Bagga, S., Goyal, A., Gupta, N., and Goyal, A. (2020). Credit card fraud detection using pipeling and ensemble learning. Procedia Computer Science, 173: 104–112. https://doi.org/10.1016/j.procs.2020.06.014
[16] Iqbal, S., Irfan, M., Ahsan, K., Hussain, M. A., Awais, M., Shiraz, M., Hamdi, M., and Alghamdi, A. (2020). A Novel Mobile Wallet Model for Elderly Using Fingerprint as Authentication Factor. IEEE Access, 8, 177405–177423. https://doi.org/10.1109/ACCESS.2020.3025429
[17] German credit data. (30 August 2021). Available: https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data)
[18] Imran, K., Anjum, N., Alghamdi, A., Shaikh, A., Hamdi, M., and Mahfooz, S. (2022). A Secure and Efficient Cluster-Based Authentication Scheme for Internet of Things (IoTs). Computers, Materials & Continua, 70(1), 1033–1052. https://doi.org/10.32604/cmc.2022.018589
[19] Khan, M. A., Quasim, M. T., Alghamdi, N. S., and Khan, M. Y. 2020. A secure framework for authentication and encryption using improved ECC for IoT-based medical sensor data. IEEE Access, 8, 52018–52027. https://doi.org/10.1109/ACCESS.2020.2980739
[20] Quasim, M. T., Khan M. A, Algarni F., Alharthy A., and Alshmrani G. M. M. (2020). Blockchain Frameworks. In: Khan M., Quasim M., Algarni F., Alharthi A. (eds) Decentralised Internet of Things. Studies in Big Data, vol 71. Springer, DOI: https://doi.org/10.1007/978-3-030-38677-1_4
[21] Khan, M. A., and Quasim, M. T. et. al., (2020). Decentralised IoT, Decenetralised IoT: A Blockchain perspective, Springer, Studies in BigData, 2020, DOI: https://doi.org/10.1007/978-3-030-38677-1

[22] Mahoto, N. A., Iftikhar, R., Shaikh, A., Asiri, Y., Alghamdi, A., and Rajab, K. (2021). An Intelligent Business Model for Product Price Prediction Using Machine Learning Approach. Intelligent Automation & Soft Computing, 29(3), 147–159. https://doi.org/10.32604/iasc.2021.018944

[23] Quasim, M. T., and Khan M. A. et.al., (2019). Internet of Things for Smart Healthcare: A Hardware Perspective, 2019 First International Conference of Intelligent Computing and Engineering (ICOICE), Hadhramout, Yemen, 2019, pp. 1–5. DOI: https://doi.org/10.1109/ICOICE48418.2019.9035175

[24] Ghazali, O., Leow, C. Y., Qaiser, S., Pattabiraman, N., Vasuthevan, S., Abdusalam, E., and Barakat, M. M. (2019). Cloud-based global online marketplaces review on trust and security. International Journal of Interactive Mobile Technologies, 13(4). https://doi.org/10.3991/ijim.v13i04.10523

[25] A. Munusamy et al., "Service Deployment Strategy for Predictive Analysis of FinTech IoT Applications in Edge Networks," in IEEE Internet of Things Journal, doi: https://doi.org/10.1109/JIOT.2021.3078148

[26] Shaikh, M., Shaikh, A., Memon, M., Shah, A., and Shah, R. (2018). State-Full Virtual Machine Live Data Migration For Improved Load Balancing. Sindh University Research Journal-SURJ (Science Series), 50(01), 107–114. https://doi.org/10.26692/Surj/2018.01.0018

[27] S. Nandy et al., "An Intrusion Detection Mechanism for Secured IoMT framework based on Swarm-Neural Network," in IEEE Journal of Biomedical and Health Informatics, doi: https://doi.org/10.1109/JBHI.2021.3101686

8 Authors

Arjwan H. Almuteer: is an IT graduated student in College of Computer, Qassim university, Buraydah, Saudi Arabia.

Asma A. Aloufi: is an IT graduated student in College of Computer, Qassim university, Buraydah, Saudi Arabia.

Wurud O. Alrashidi: is an IT graduated student in College of Computer, Qassim university, Buraydah, Saudi Arabia.

Jowharah F. Alshobaili: received her B.Sc. degree in Computer Science, MSc degree in Artificial Intelligence, specifically in Bioinformatics. At present, she is a Lecturer and coordinator at the Department of Information Technology, College of Computer, Qassim University, Buraydah, Saudi Arabia. E-mail: j.alshobaili@qu.edu.sa.

Dina M. Ibrahim: is an assistant professor at the department of Information Technology, College of Computer, Qassim University, Buraydah, Saudi Arabia from September 2015 till now. She was born in the United Arab of Emirates, her B.Sc., M.Sc., and Ph.D. degrees taken from the Computers and Control Engineering Department-Faculty of Engineering, Tanta University in 2002, 2008, and 2014, respectively. Her research interests include networking, wireless communications, machine learning, security, and the Internet of Things. Dina has published more than 42 articles in various refereed international journals and conferences. She is serving as a reviewer in Wireless Network (WINE) the Journal of Mobile Communication, Computation, and Information since 2015 Dina also acts as a Co-Chair of the International Technical Committee for the Middle East Region of the ICCMIT conference since 2020. E-mail: d.hussein@qu.edu.sa, dina.mahmoud@f-eng.tanta.edu.eg.

Article submitted 2021-09-06. Resubmitted 2021-10-19. Final acceptance 2021-10-19. Final version published as submitted by the authors.