A Financial Fraud Detection Model Based on LSTM Deep Learning Technique A Financial Fraud Detection Model Based on LSTM Deep Learning Technique

Article *in* Journal of Applied Security Research · September 2020

DOI: 10.1080/19361610.2020.1815491

CITATIONS
110

READS
5,619

3 authors, including:

Murad A. Rassam
Taiz University
61 PUBLICATIONS **1,330** CITATIONS

SEE PROFILE

Routledge
Taylor & Francis Group

Journal of Applied Security Research

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/wasr20

A Financial Fraud Detection Model Based on LSTM Deep Learning Technique

Yara Alghofaili , Albatul Albattah & Murad A. Rassam

To cite this article: Yara Alghofaili , Albatul Albattah & Murad A. Rassam (2020): A Financial Fraud Detection Model Based on LSTM Deep Learning Technique, Journal of Applied Security Research, DOI: 10.1080/19361610.2020.1815491

To link to this article: https://doi.org/10.1080/19361610.2020.1815491

Published online: 10 Sep 2020.

JOURNAL OF APPLIED SECURITY RESEARCH
https://doi.org/10.1080/19361610.2020.1815491

A Financial Fraud Detection Model Based on LSTM Deep Learning Technique

Yara Alghofaili[a], Albatul Albattah[a], and Murad A. Rassam[a,b]

[a]Department of Information Technology, College of Computer, Qassim University, Buraydah, Saudi Arabia; [b]Faculty of Engineering and Information Technology, Taiz University, Taiz, Republic of Yemen

ABSTRACT

As the use of the internet is growing exponentially, more and more businesses such as the financial sector are operationalizing their services online. Consequently, financial frauds are increasing in number and forms around the world, which results in tremendous financial losses which make financial fraud a major problem. Unauthorized access and irregular attacks are examples of threats that should be detected by means of financial fraud detection systems. Machine learning and data mining techniques have been widely used to tackle this issue over the past few years. However, these methods still need to be improved in terms of speed computation, dealing with big data, and identify the unknown attack patterns. Therefore, in this paper, a deep learning-based method is proposed for the detection of financial fraud based on the Long Short-Term Memory (LSTM) technique. This model is aimed at enhancing the current detection techniques as well as enhancing the detection accuracy in the light of big data. To evaluate the proposed model, a real dataset of credit card frauds is utilized and the results are compared with an existing deep learning model named Auto-encoder model and some other machine learning techniques. The experimental results illustrated a prefect performance of LSTM where it achieved 99.95% of accuracy in less than a minute.

KEYWORDS

Financial fraud; fraud detection; deep learning; long short-term memory

Introduction

In the last decade, there was an exponential growth of internet users, online services, and business. Today people pay all their bills using online payment services; they shop and pay using credit or debit cards and even people can get and transfer funds using their online banking systems. All this made our lives easier; people do not have to stand in long queues for paying electricity bills, and they do not have to carry a lot of cash when going to the market for shopping, etc. Despite all these positive aspects,

CONTACT Murad A. Rassam m.qasem@qu.edu.sa Department of Information Technology, College of Computer, Qassim University, Buraydah, Saudi Arabia.

online transactions brought the great danger of unauthorized payments, which are known as financial frauds. These can be banking transaction fraud, online identity theft, insurance frauds, payment card frauds, and money laundering (Yomas & Kiran, 2018).

The fraudulent financial activities are very sophisticated and very complicated to identify. Frauds are increasing significantly with the development of modern technologies, particularly in the financial sector (Tripathi & Pavaskar, 2012).There are several forms of frauds in financial systems such as online banking fraud, credit card fraud, fraudulent loans, falsification of documents, Phishing, scamming, fraudulent accounts among others. Fraud crimes cost financial establishments millions of dollars yearly, which impacts the establishment's financial situation and the confidence of customers (West & Bhattacharya, 2016).

The global financial institutions and companies are experiencing massive losses due to various multiple financial frauds. Every day there are news that credits details are stolen, an unauthorized transaction using credit and debit card details without the knowledge of the real user, is sending alarm bells for the banking sector, customers and, governments around the world.

Financial fraud became a massive problem. Unauthorized access and unusual attacks are identified using system for detecting financial fraud. Financial institutions must constantly develop their mechanisms for detecting fraud. Machine learning and data mining strategies are now commonly used over the last few years to fix this problem. Many of existing studies (Meng, et al., 2018; Modi & Dayma, 2017; Sadgali et al., 2019; Zhang & Trubey, 2019) used machine learning and data mining techniques to detect financial frauds. However, these techniques still need to be improved in terms of computational cost, memory cost and dealing with big data which becomes a feature of current financial transactions. The detection of financial fraud is a challenging problem due to four major reasons which are (1) fraudulent behavior is constantly changing, (2) there is no mechanism to track the information about the fraud transaction, (3) the existing detection techniques (like machine learning algorithms) have certain limitations, and (4) financial fraud datasets are highly skewed, therefore it is hard to train algorithms.

The contribution of this paper is to propose a fraud detection model based on deep learning-based technique (Long Short – Term Memory). This model is aimed to identify the suspicious financial transaction and alert the relevant authorities about it, to take appropriate action. As a result, the proposed model may constitute a useful tool for the financial sectors to reduce their potential losses.

The rest of this paper is structured as follows: related fraud detection models in financial sector are presented in Section "Related words."

The proposed model is described in Section "Proposed model." The experimental investigation, results analysis, and evaluation of the proposed model are reported in Section "Experimental Investigation." Section "Result and Analysis" reports a comparison with existing models and Section "Conclusion and Future Work" presents the conclusion and future research directions.

Related works

The global financial services and businesses suffer huge frauds that have led to the overthrow of entire organizations, large investment losses and significant legal costs. For this reason, the investors and academic researchers have received considerable interest to detect fraud in the financial domain (Sharma & Panigrahi, 2013).

Fraud detection is an important process in financial activities. A lot of attention has been given to fraud detection techniques in the last decade. Several works exploited the advantages of machine learning and deep learning algorithms to detect and predict fraudulent financial activities. For example, a study in (Patidar & Sharma, 2011) applied neural networks to solve the fraud detection problems in credit cards.

Another study in (Sahin & Duman, 2011) compared the performance of Artificial Neural Network and Logistic Regression in credit card fraud detection based on a real dataset. The empirical results of the study illustrated an equal performance of these models over training data. Also, it reveals the advantage of Artificial Neural Network over Logistic Regression on the test data. However, Artificial Neural network data need training to expect output. Thus, the Artificial Neural network uses for classification tasks not for detection fraud or abnormal behavior detection tasks.

The authors in (Jha et al., 2012) used a Logistic Regression to show the enhanced performance of a transaction gathering strategy to create appropriate derived attributes, that used to detect fraud in credit cards. Based on real-world data from credit card transactions, they developed a method that may improve fraud detection in credit cards. The results illustrated the importance of classification of features like product types, transaction types, and location, … etc to identify credit card frauds. However, use of a Logistic Regression algorithm to detect fraud is not a good idea because it can expect only a categorical outcome. Moreover, this algorithm is also known for its vulnerability to over-fitting (Patil et al., 2018).

The study in (Zareapoor et al., 2012) compared nine fraud detection methods on credit cards which are Decision Tree, Neural Network, Bayesian Network, support vector machine, genetic algorithm, k nearest neighbor and Artificial Immune System, Hidden Markov Model, fuzzy

neural network and fuzzy Darwinian system. The Comparison used metrics such as accuracy, speed, and cost to evaluate the performance of the methods. As a result, Fuzzy Darwinian method achieved very high accuracy and HMM achieved quick processing speed. Therefore, to obtain a better solution to build a hybrid approach to develop some efficient algorithms that can work well with variable misclassification costs and greater accuracy for the classification problem. Although Fuzzy Darwinian has a high performance, it has very low speed in detection and high cost. Also, HMM achieved quick processing speed, but it not scalable to large data, high cost, and has low accuracy (Sorournejad et al., 2016).

A Study in (Olszewski, 2014) proposed a fraud detection approach based on Self-Organizing Map (SOM) neural network. This approach utilized user accounts visualization and type of threshold detection. The empirical study that was applied on a real dataset confirms the data visualization advantage, which converts the input of high-dimensional data into a 2-dimensional image which makes more sense to a common person. The SOM algorithm suffer some restrictions such as slow computation and lack of parallelism capability for big datasets (Fort, 2002).

The authors in (Agrawal & Agrawal, 2015) presented different approaches for anomaly detection such as Clustering-based Anomaly Detection techniques, Classification based anomaly detection, and Hybrid approaches. Hybrid approaches gained better results and overcame the disadvantages of other approaches. Nevertheless, these approaches take long time in training, high cost for computational resources, and have complex architecture (Kumar et al., 2020).

Authors in (Bhusari & Patil, 2016) discussed the Hidden Markov Model that may help to find out the fraudulent transaction, which contains different ranges of the transaction such as low, medium and high as the observation symbols. The Hidden Markov model made the fraud detection systems very simple not taking a long time although having complex processes. Usually Hidden Markov Model required training by use of annotated data. In addition to, this model needs manual markup. Therefore, this model not suitable for detect unknown patterns and not effective to identify new patterns of fraud.

A survey by (Ahmed et al., 2016) evaluated various clustering-based fraud detection methods in financial events. A study by (Modi & Dayma, 2017) compared various methods for detecting fraudulent transactions which are Artificial Neural Network, Hidden Markov Model, Decision tree, Rule-based methods and Convolutional Neural Network. These methods can be used to detect fraudulent transactions individually or in a group. Nevertheless, theses algorithms have some drawbacks such as, need excessive training, and requirement to test each condition one by one. In fraud

detection condition is transactions, therefore, it will take high processing time (Sorournejad et al., 2016).

Moreover, the authors in (Awoyemi, 2017) compared in the performance of three techniques naïve Bayes, k-nearest neighbor and logistic regression that applied on data of credit card. The results showed an accuracy of 97.92, 97.69, and 54.86% for naïve Bayes, k-nearest neighbor, and logistic regression. A comparative study showed that k-nearest neighbors perform better than naïve bays and logistic regression techniques. However, k-nearest neighbor need data training to classify. As a result, this algorithm is not effective to detect anomaly behavior because it needs to be trained before on known patterns.

These algorithms were also used for identifying suspicious patterns in transactions. Once a suspicious transaction is identified, it is blocked or quarantined. The work of (Zhang & Trubey, 2019) critically analyzed the interplay of machine learning and sampling schemes in an experiential analysis by using actual transaction data. This study includes five major machine learning algorithms which are Random Forest, Artificial Neural Network, Decision tree, Logistic regression, and Naïve Bayes. This work provides insights into the use of machine learning in identifying suspicious activities and classification of rare events in general. However, these algorithms need to train the data to predict the output, therefore not suitable for detect the new patterns. Moreover, they need to large space of memory when dealing with big data (Bhavsar & Ganatra, 2012).

Authors in (Sadgali et al., 2019) studied a state-of-the-art techniques that can detect different frauds. The techniques such as classification, clustering, and regression were investigated to identify the contribution of each technique and its effectiveness. According to authors, machine-learning techniques have an essential role in fraud detection and being applied to extract and uncover the hidden data. Furthermore, the hybrid fraud detection techniques were the most applicable as a result of integrating the strengths of several traditional techniques of detection. However, the most of hybrid techniques did not work in real time.

Intrusion detection systems play a very critical role in financial fraud detection. These systems use to identify suspicious groups. The study in (Sasirekha, 2012) proposed an intrusion detection system for detecting credit card fraud by integrating three methods anomaly, misuse, and decision-making models. The anomaly detection method is implemented by using Hidden Markov which identifies the credit card transaction as malignant or benign based on the credit card user's expenditure profile. The behaviors of the credit card holder are taken as attributes, and the user's spending profile detects the anomalous transactions. The transactions considered abnormal or suspicious are then submitted to the monitoring

system for misuse. Thus, the transactions are compared to predefined forms of attack and then sent to the model of decision making to identify it as a known or unknown form of attack. The decision-making model is then used to interpret the findings observed and monitor the forms of attacks in the credit card system. However, this system needs to train to detect fraud. Moreover, the Hidden Markov model has many limitations like low accuracy, high cost, and not suitable for big data.

Authors in (Sureshkumar, 2019) discussed issues of current techniques used in fraud detection in many areas such as credit card and intrusion detection. The result showed the effectiveness of these techniques in some kinds of fraud. Nevertheless, still have some problems like fraud detection for a credit card have a few approaches are available in public. Also, intrusion detection is difficult to test and has poor portability because the rules and systems must be specified to the environment being monitored. Besides, such systems need updating to keep them up-to-date with existing methods of fraud.

Deep learning technologies are the most up-to-date technologies used to classify data in a different perspective. Many researchers used them to detect frauds on credit cards that cannot be detected based on supervised machine learning or on the details of the previous history. In this perspective, a study in (Pandey, 2017) used deep learning to understand how can detect credit card frauds. In their study, H2O deep learning framework was introduced. This framework aimed to handle massive datasets by Calculating the Mean Squared Errors, Root Mean Squared Errors, Mean Absolute Errors, and Root Mean Squared Log Errors. The results showed that the Mean Squared Error = 0.01661, Root Mean Squared Errors = 0.1288, Mean Absolute Errors = 0.03198, and Root Mean Squared Log Errors = 0.09009 in detecting fraudulent transactions.

Another study in (Pumsirirat & Yan, 2018) proposed a deep learning model based on auto-encoder (AE) technique to detect a credit card fraud. This model applied backpropagation by setting the inputs equal to the outputs. The restricted Boltzmann machine (RBM) includes two layers, the first layer is input visible and the other is the hidden layer. The results illustrated 96.03% of accuracy.

In the study of (Roy, 2018), the effectiveness of deep learning techniques was evaluated on a dataset of approximately 80 million transactions of credit cards that have been pre-identified as legitimate and fraudulent. This study compared the performance of some deep learning algorithms in terms of class imbalanced, sensitively analysis of the parameters, and scalability. Many techniques such as Gated Recurrent Unit, Long Short-term Memory, Recurrent Neural Networks and Artificial Neural Networks were used in distribute cloud computing environment. The results showed that

the Long Short-term Memory technique achieved the best performance. Also, it can deal with big complex datasets.

Recently, authors in (Kim et al., 2019) developed champion-challenger framework to compare between the deep learning neural network model and the hybrid ensemble of different machine learning model to determine which was performed better in fraud detection in term of the off-line and post-launch evaluation. The results showed that the deep learning model was more effective than the hybrid ensemble model.

To conclude, research and development work in the area of financial fraud detection focused on the use of machine learning algorithms such as Support Vector Machine, Decision tree, Logistic regression, and Naïve Bayes … etc. These algorithms suffer certain limitations such as ineffectiveness when dealing with complex non-linear data, slow computation, ineffective memory utilization, and lack of parallelism.

Therefore, deep learning is a perfect candidate to overcome the above-stated limitations of classical machine learning algorithms. Deep Learning enables computational models consisting of several layers of processing to learn data representation at various abstraction levels. These approaches have significantly enhanced the state-of-the-art object recognition, object detection and voice recognition (LeCun et al., 2015). In addition, the architecture of deep learning models is dynamic to adapt to new patterns. Furthermore, it can process big data efficiently in terms of accuracy, memory and speed. Thus, its performance is better than machine learning on huge amount of data (Alom et al., 2019), as shown in Figure 1.

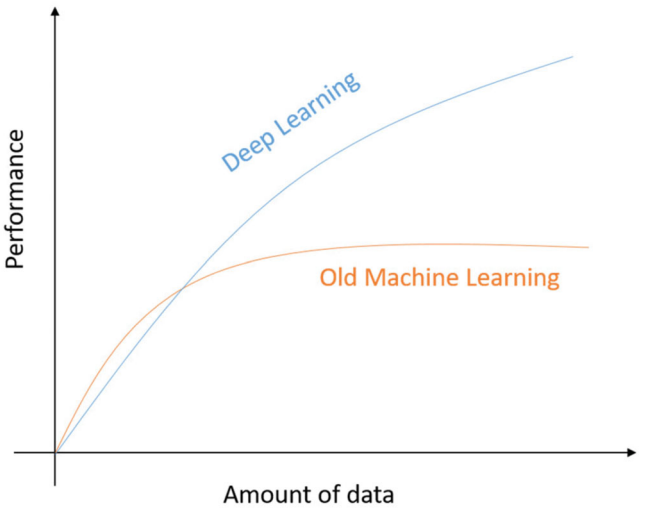

Figure 1. The performance of deep learning compared to old machine learning (Alom et al., 2019).

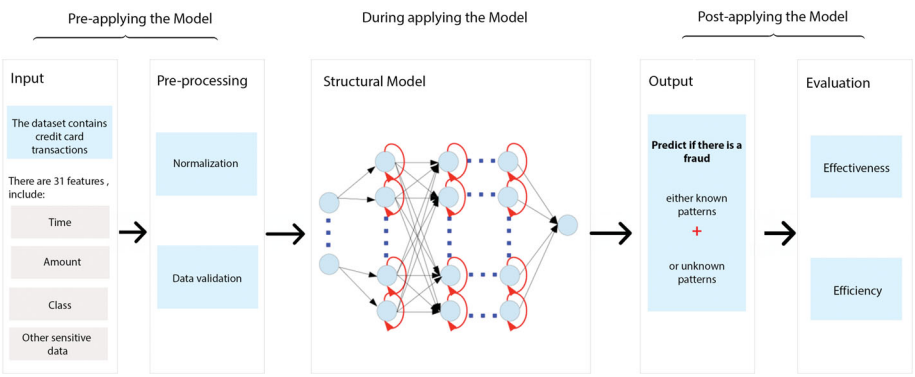

Figure 2. Proposed model.

Proposed model

This section introduces the proposed financial fraud detection model and describe its stages. The model is divided into 3 stages namely pre-applying the model stage, during applying the model stage and post-applying the model stage. Figure 2 shows the stages which are described in details in the subsequent subsections.

Pre-applying the model

This stage constitutes substages that start with the input sub-stage which contains the preparation of the dataset samples to train and test the model. The dataset contains credit-card transactions which happened within two days, there exist 492 frauds out of 284,807 transactions. As a result, the dataset is strongly unbalanced, accounting for 0.172% of all transactions in the positive class (fraudulent activity). Besides, this dataset has 31 features that includes time, amount, class, and other sensitive features. It has been used as a benchmark dataset by various researchers (Carcillo et al., 2018, 2019; Dal Pozzolo et al., 2014). Before applying the model on the dataset, three procedures should be applied which are data validation, normalization and dividing.

a. Data validation
   This step is applied to check the validity of the data inside the dataset like a negative value of time, empty values, or negative amount.
b. Normalization
   To have an accurate result, the model rescales the variables to be in the range between -1 to 1. This step is necessary to change the numeric column values in the dataset to be used on a common scale, without deforming variation in the ranges of values or losing data. The normalization procedure uses Equation 1.

$$x(i) = \frac{x(i) - \overline{x}}{s(x)} \tag{1}$$

c. Dataset samples divide
Dividing the data samples into training and testing is essential for getting a realistic assessment of the performance. The proposed model has allocated 70% of the dataset for the training and the residual 30% for testing.

Applying the model

During this stage, two substages are involved in order to apply the proposed LSTM which constitutes the heart of the proposed deep learning model. The substages are the creation of the LSTM structure and the setting up its parameters.

Creating the LSTM structure

The proposed deep learning financial fraud detection model in this paper relies on Long Short-Term Memory (LSTM) technique, an extension for Recurrent neural networks (RNNs) which are forms of deep learning neural networks with memory. It is well suited for prediction, due to prior knowledge and the relation between prediction results and historical input data. The LSTM architecture allows learning through long dependency on sequences prediction issues. It is useful for longer-term patterns and it can maintain a long memory (Malhotra, 2015). Besides, it has default behavior which keeps information for a long-term period. Therefore, it is a perfect tool for prediction and detection tasks according to authors in (Islam et al., 2019). In the applying model stage, the prepared data in the pre-applying model stage is fed into the model and processed by the layers that contain LSTM cells. The structure of the LSTM cell is shown in Figure 3.

As shown in Figure 3, the structure of The LSTM network includes memory blocks (cells) which have several states and gates. The cell state is the major chain of information flow. It allows the information to flow forward unaltered. The forget gate (f_t) determines what information must be removed or kept. Data from the prior hidden state (h_{t-1}) is passed via the sigmoid function together with data from the current input (X_t). The sigmoid function (σ) determines values that are between 0 and 1; if the value is nearer to 0 that is meant to forget, and if the value is nearer to 1 means to keep. Also, cell state vector C_{t-1} control which elements that will forget. The Equation 2 explains how forget gate is calculated:

$$\text{ft} = \sigma\left(Wf\ [h_{t-1},\ X_t] + b_f\right) \tag{2}$$

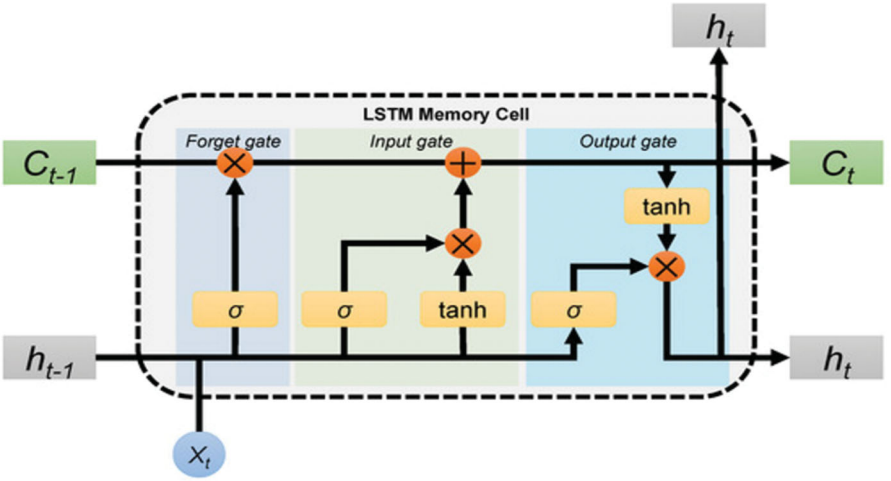

Figure 3. LSTM cell structure (Fan et al., 2020).

Where b_f and W_f denote the biases and input weights matrices of forget gate (f_t).

The input gate (I_t) determines which data is necessary to add from the current input (X_t) and also to update cell state. This gate used the *tanh* function (N_t) to pass the current input and the hidden state to make values between -1 and 1 to assist regulate the network. Moreover, to generate new memory, it is then applied to the old cell-stated (C_{t-1}) memory at time $t - 1$, which produce new cell-stated (C_t) at time t as in Equations 3–5.

$$\mathrm{It} = \sigma\left(Wi\left[h_{t-1}, X_t\right] + bi\right) \tag{3}$$

$$Nt = \tanh\left(Wn\left[h_{t-1}, X_t\right] + bn\right) \tag{4}$$

$$Ct = C_{t-1} * ft + Nt * It \tag{5}$$

Where b and W denote the biases and input weights matrices of input gate (f_t).

The output gate (O_t) conditionally determines what to output of the next hidden state (h_t) through the output of the sigmoid gate and with new values generated by *tanh* from cell state as in Equations 6–7.

$$Ot = \sigma\left(Wo\left[h_{t-1}, X_t\right] + bo\right) \tag{6}$$

$$ht = Ottanh\left(Ct\right) \tag{7}$$

Where b and W denote the biases and input weights matrices of output gate (f_t).

Table 1. LSTM parameters and performance measures.

Parameters	Description
Optimizer	Parameter that work to enhance the performance. It has different types such as, Adam, RMSprop … etc.
Metric	Measure the performance like , accuracy … etc.
Batch size	Refers to the number of windows of data we are passing at once.
Epochs	Refers to the number of iterations (forward and back propagation) model needs to make.

Table 2. Optimizers of LSTM model.

Optimizers	Description
RMSprop	Root Mean Square Propagation (RMSprop) proposed by Geoff Hinton. It is an unpublished optimization technique for gradient descent, designed for artificial neural networks. RMSprop is a popular choice for an adaptive learning rate technique (Ruder 2016).
Adam	Adaptive Moment Estimation (ADAM) adaptively adjusts the learning rates computation for every parameter for deep learning artificial neural network algorithms. To achieve this goal, it uses the first and second moment of the gradient. Also, ADAM is useful in the problems that consist of large amount of data (Ruder 2016).
Adagrad	Adaptive Gradient (Adagrad) is a gradient-based optimization technique. Adagrad supports the adaptive learning rate and tunes the parameters accordingly. It performs smaller updates. Due to this characteristic, it is useful in the problems that consist of sparse data (Ruder 2016).

LSTM parameters and performance measures

The model uses several parameters called hyperparameters that work to improve the results. To measure the performance of the proposed model, four parameters namely optimizer, batch size, epochs, and matrix are used and described in Table 1. There are a considerable number of optimizers that could make the difference between models, some of these optimizers are shown in Table 2.

Post-applying the model

This step contains two-part, which are the obtain of output and evaluation of the output. The output for the dataset will be shown as well as will decide if there is fraud or not. Then, the results obtained will be analyzed and evaluate these results. The evaluation process is based on adapting different parameters of the model, tested on a different number of layers, and a different number of iterations. The goal is to find out which adaptive parameters provide the best accuracy of prediction and execution time. As well as test what are the number of layers and the number of iterations that will provide the best results.

Experimental investigation

This section discusses about the dataset and model setup.

Credit card fraud detection dataset

The dataset used in this study as described in Section "Pre-applying the model" contains numeric columns called features.These features contain different data, for example:

1. "Time" refers to the seconds between the transactions for the same credit card.
2. "Amount" refers to the state of amounts in the transaction.
3. "V1 to V28" some sensitive features.
4. "Class" refers to the fraudulent activity if column =1, it means there is a fraud, but if column = 0, it means there is no fraud.

Model setup

In this section, the model setup is conducted to give convincing pieces of evidence that the LSTM model is useful in the detection of anomalies in the finance domain. Functions from the Sklearn library of Python is used for programming the different tasks, such as pre-processing and model selection. Keras library is used for the implementation of the LSTM model. The aim is to measure the accuracy, loss rates, and execution time for financial detection fraud. Also, it shows that the LSTM model can detect known and unknown patterns of fraud.

Accuracy

The accuracy metric is a statistical calculation of how a model predicts correctly (Ahmed & Bahador, 2018). The aim of calculating the accuracy is to achieve the model's efficiency. Equation 8 explains how accuracy metric is calculated:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{8}$$

Where TP denote true positive, which indicates a part of the suspicious transactions properly reported as suspicious, TN denote true negative, which indicates a part of the normal transactions properly classified as normal, FP denote false positive, which represents the portion of the non-suspicious transactions wrongly identified as suspicious transactions, and FN denote false negative, which represents the part of the suspicious transactions wrongly being identified as normal transactions (Sorournejad et al., 2016).

Loss rate

Loss rate is a function that measures the difference between the actual output and predicted output during training to speed the process of learning (Ahmed & Bahador, 2018). Also, loss rate is used to evaluate the model

Table 3. Initial experimental result.

Optimizer	Training Accuracy (%)	Validation Accuracy (%)	Loss (%)	Number of Layers	Number of iterations	Time(s)
Adagrad	99.90	99.93	0.46	3	100	307s

performance and minimizing the error. The loss rate can be calculated as in Equation 9.

$$Loss = \ Y - \ Log(Y_{Pred}) + (1 - Y) - \ Log(1 - Y_{Pred}) \tag{9}$$

Where Y denote actual output and Y_{Pred} denote predict output.

Execution time

The execution time is measured as the period that the model has spent performing the task. The aim of calculating the time of execution is to know how much time the model takes to detect the frauds as well as making sure that the model achieves the its goal efficiently. Equation 10 explains how execution time is calculated.

$$Time = \ Time_{end} - \ Time_{start} \tag{10}$$

Result and analysis

In this section, the results of applying the proposed model are explored. Furthermore, an analysis of the reported results is also conducted.

Experimental result

As an initial result, the model obtained 99.90% of accuracy and 0.46% of loss rate in 307 s. This experiment used Adagrad optimizer, 100 times of iterations, and 3 layers for encoding and decoding as shown in Table 3.

To evaluate the proposed model performance in terms of accuracy and speed, it is executed using different settings such as different optimizers, different numbers of layers, and iterations. Several optimizers like Adam, Adagrad, and RMSprop have been used in the evaluation; each optimizer gives a different result. The experiments aimed to determine which optimizer obtaines the best performance. The results of different experiments showed that Adam optimizer obtained optimum accuracy as shown in the Table 4. It is known that Adam optimizer can deal with big data, does not require a large memory space, and is computationally efficient (Kingma & Ba, 2015).

Based on the results shown in Table 4, Adam optimizer achieved good results, therefore, it was further evaluated on several layers and a different

Table 4. Result of experiment with different optimizers.

Optimizer	Training Accuracy (%)	Validation Accuracy (%)	Loss (%)	Number of layers	Number of iterations	Time(s)
Adagrad	99.90	99.93	0.46	3	100	307s
RMSprop	99.94	99.94	0.35	3	100	385s
Adam	99.96	99.96	0.21	3	100	405s

Table 5. Adam optimizer result with different number of layers.

Training accuracy (%)	Validation accuracy (%)	Loss (%)	Number of layers	Number of iterations	Time(s)
99.96	99.96	0.22	2	100	400s
99.96	99.96	0.21	3	100	405s
99.96	99.96	0.22	5	100	468s

Table 6. Adam optimizer result with different number of iterations.

Training accuracy (%)	Validation accuracy (%)	Loss (%)	Number of layers	Number of iterations	Time(s)
99.95	99.95	0.31	3	10	45s
99.96	99.96	0.21	3	100	405s
99.9%	99.96	0.18	3	200	868s

number of iterations. The number of layers has not achieved a significant change in accuracy as it illustrates in Table 5. Also, the number of iterations has not changed much in results, where the 10 times of iterations achieved 99.95% of accuracy and 0.22% loss rate, while the 100 times of iterations achieved 99.96% accuracy and 0.21% loss rate. Also,the 200 times of iterations achieved 99.97% accuracy and 0.22% loss rate. As shown in Table 6.

Results analysis

The model efficiently and effectively predicts financial fraud based on applying the LSTM cells. There are a large number of optimizers that can be used in this model, however, selecting the optimum optimizer brings a significant improvement in results. In the experiment, the number of layers and also the number of iterations did not make much difference in the results. Also, the results point out that time is affected by the number of iterations and the number of layers. The results proved that therein a strong correlation between accuracy and the number of iterations. Besides, there is an inverse relationship between loss rates and the number of iterations. There exists a negative correlation between accuracy and loss rates as when accuracy increases the loss rates decreases, as shown in Figure 4.

Comparison with other deep learning based technique

To show the effectiveness of the proposed model, it is compared with the Autoencoder model. Through experiments, Adam optimizer shows high accuracy and speed over the other optimizers, therefore, this comparison is used by Adam optimizer with 3 layers and 100 iterations. As the results

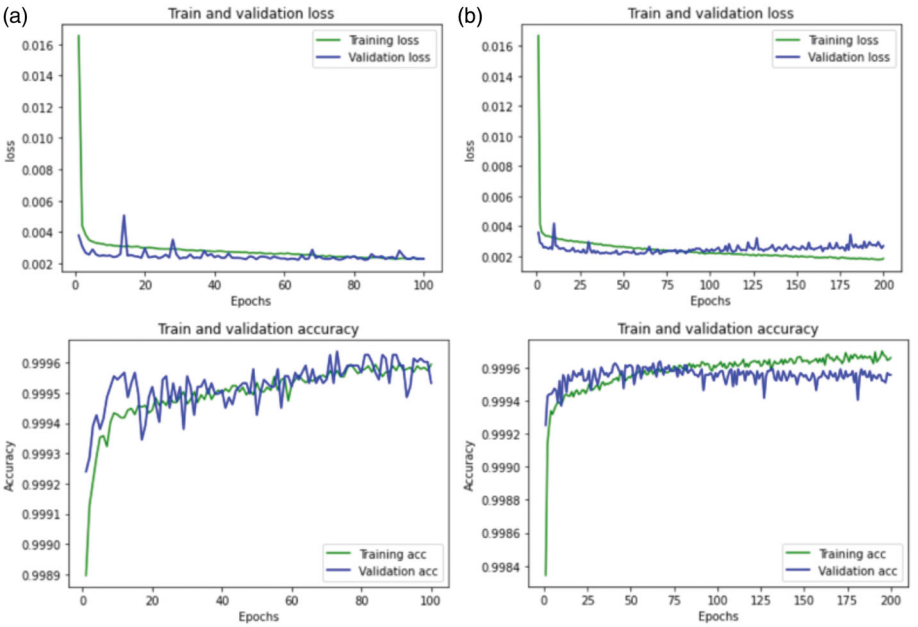

Figure 4. Results (a) showed the results of 100 iterations loss and accuracy (b) showed the results of 200 iterations loss and accuracy.

Table 7. Comparison with auto-endorer.

Model	Optimizer	Training accuracy (%)	Validation accuracy (%)	Loss (%)	Number of layers	Number of iterations	Time(s)
LSTM	Adam	99.96	99.96	0.21	3	100	405s
Auto-encoder	Adam	70.27	69.95	69.08	3	100	318s

shown in the Table 7, the LSTM model outperformed the Auto-encoder model. LSTM model obtained 99.96% of accuracy and 0.21% loss rate in 405 s.Whereas, Autoencoder model obtained 70.27% of accuracy and 96.08% loss rate in 318 s. The result proved that the LSTM model can learn even through complex data patterns. Also, the LSTM model is able to deal better with big data.

Comparison with existing machine learning based techniques

The LSTM based financial fraud detection model was compared with existing machine learning based models like, Random Forest (RF), Logistic Regression (LR), and Support Vector Machine (SVM). These machine learning techniques were applied to the same data set and the results are shown in Table 8. RF obtained the same accuracy of LSTM in 10 times of alteration. But RF took a longer time in execution and required advanced training of data to predict the output (Patil et al., 2018). Therefore, it was not effective to detect new patterns of frauds. Whereas SVM obtained

Table 8. Comparison with existing machine learning techniques.

Model	Accuracy (%)	Time (s)	Drawbacks
RF	99.95	200s	• RF can achieve good results with small number of data. • RF quickly reaches a point that can't enhance the accuracy. • Need to train the data to predict the results.
SVM	99.87	435s	• SVM is not appropriate for big and complex data. • SVM required training by use of annotated data which means not effective to identify new patterns of fraud. • SVM has lack of results transparency.
LR	99.88	10s	• Logistic Regression it can expect only a categorical outcome. • Its vulnerability to over-fitting.

99.87% of accuracy in 435 s. SVM obtained less results in terms of speed and accuracy (Sorournejad et al., 2016). Therefore, it is not suitable for big and complex data. Besides, LR achieved 99.88% of accuracy in 10 s. Although, LR took a shorter time but it obtained less accuracy than LSTM. In addition, LR can expect only a categorical outcome according to (Patil et al., 2018). Despite some machine learning algorithms have shown good results, but they are unable to predict new patterns or deal with big data because they may reach a point at which they cannot enhance accuracy. However, deep learning-based methods can learn even through complex data patterns and dynamically adapt with new patterns of frauds. Table 8 further shows the drawbacks of each machine learning techniques in the context of financial fraud detection problem.

Conclusion and future work

Financial fraud is a problem with far-reaching implications for the financial sector and stakeholders. Increased reliance on emerging technology has compounded the issue in recent years. Traditional approaches are ineffective in the big data age. Therefore, the work developed a model for the detection of financial fraud based on the Long Short-Term Memory (LSTM) technique using a real data set of credit card fraud. This model was aimed to enhancing the current detection techniques as well as enhancing the detection accuracy in the light of big data. It addressed the problem of detection of unknown and sophisticated patterns of fraud by using deep learning techniques to identify patterns quickly and with high accuracy. Also, the problem of the inefficiency of the existing techniques was addressed by using the proposed model based on the LSTM technique. Finally, a comparison with Auto-encoder and other existing machine learning techniques showed that the LSTM technique can achieve perfect performance in addressing fraud detection problems. As future work, an

algorithm can be developed to perform various tasks like, calculate the timing of the fraud that occurred in addition to the location of the fraud.

ORCID

Murad A. Rassam http://orcid.org/0000-0003-3558-6737

References

Agrawal, S., & Agrawal, J. (2015). Survey on anomaly detection using data mining techniques. *Procedia Computer Science*, *60*, 708–713. https://doi.org/10.1016/j.procs.2015.08.220

Ahmed, M., Mahmood, A. N., & Islam, M. R. (2016). A survey of anomaly detection techniques in financial domain. *Future Generation Computer Systems*, *55*, 278–288. https://doi.org/10.1016/j.future.2015.01.001

Ahmed, W., & Bahador, M. (2018). *The accuracy of the LSTM model for predicting the S&P 500 index and the difference between prediction and backtesting.* (Degree thesis). Stockholm, Sweden: KTH Royal Institute of Technology, School Of Electrical Engineering and Computer Science.

Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S., Hasan, M., Van Essen, B. C., Awwal, A. A. S., & Asari, V. K. (2019). A state-of-the-art survey on deep learning theory and architectures. Electronics, *8*(3), 292. https://doi.org/10.3390/electronics8030292

Awoyemi, J. O. (2017). Credit card fraud detection using machine learning techniques: A comparative analysis. *Conference on Computing Networking and Informatics (ICCNI), 2017 International, IEEE*. https://doi.org/10.1109/ICCNI.2017.8123782

Bhavsar, H., & Ganatra, A. (2012). A comparative study of training algorithms for supervised machine learning. *International Journal of Soft Computing and Engineering (IJSCE)*, *2*(4), 2231–2307.

Bhusari, V., & Patil, S. (2016). Study of hidden markov model in credit card fraudulent detection. ld *Conference on Futuristic Trends in Research and Innovation for Social Welfare (Startup Conclave)., 2016 Wor, IEEE.* https://doi.org/10.1109/STARTUP.2016.7583942

Carcillo, F., Dal Pozzolo, A., Le Borgne, Y.-A., Caelen, O., Mazzer, Y., & Bontempi, G. (2018). Scarff: A scalable framework for streaming credit card fraud detection with spark. *Information Fusion*, *41*, 182–194. https://doi.org/10.1016/j.inffus.2017.09.005

Carcillo, F., Le Borgne, Y.-A., Caelen, O., Kessaci, Y., Oblé, F., & Bontempi, G. (2019). Combining unsupervised and supervised learning in credit card fraud detection. *Information Sciences*. https://doi.org/10.1016/j.ins.2019.05.042

Dal Pozzolo, A., Caelen, O., Le Borgne, Y.-A., Waterschoot, S., & Bontempi, G. (2014). Learned lessons in credit card fraud detection from a practitioner perspective. *Expert Systems with Applications*, *41*(10), 4915–4928. https://doi.org/10.1016/j.eswa.2014.02.026

Fan, H., Jiang, M., Xu, L., Zhu, H., Cheng, J., & Jiang, J. (2020). Comparison of long short term memory networks and the hydrological model in runoff simulation. Water, *12*(1), 175. https://doi.org/10.3390/w12010175

Fort, J.-C. (2002). *Advantages and drawbacks of the Batch Kohonen algorithm*. ESANN.

Islam, M. S., Sharmin Mousumi, S. S., Abujar, S., & Hossain, S. A. (2019). Sequence-to-sequence Bangla sentence generation with LSTM Recurrent Neural Networks. *Procedia Computer Science*, *152*, 51–58. https://doi.org/10.1016/j.procs.2019.05.026

Jha, S., Guillen, M., & Christopher Westland, J. (2012). Employing transaction aggregation strategy to detect credit card fraud. *Expert Systems with Applications*, *39*(16), 12650–12657. https://doi.org/10.1016/j.eswa.2012.05.018

Kim, E., Lee, J., Shin, H., Yang, H., Cho, S., Nam, S.-K., Song, Y., Yoon, J.-A., & Kim, J.-I. (2019). Champion-challenger analysis for credit card fraud detection: Hybrid ensemble and deep learning. *Expert Systems with Applications*, *128*, 214–224. https://doi.org/10.1016/j.eswa.2019.03.042

Kingma, D. P. & Ba, J. (2015, May 7-9). Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015*, San Diego, CA.

Kumar, P., Lai, S. H., Wong, J. K., Mohd, N. S., Kamal, M. R., Afan, H. A., Ahmed, A. N., Sherif, M., Sefelnasr, A., & El-Shafie, A. (2020). Review of nitrogen compounds prediction in water bodies using artificial neural networks and other models. Sustainability, *12*(11), 4359. https://doi.org/10.3390/su12114359

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Malhotra, P. (2015). Long short term memory networks for anomaly detection in time series. *Proceedings, Presses universitaires de Louvain.*

Meng, W., Wang, Y., Wong, D. S., Wen, S., & Xiang, Y. (2018). TouchWB: Touch behavioral user authentication based on web browsing on smartphones. *Journal of Network and Computer Applications*, *117*, 1–9. https://doi.org/10.1016/j.jnca.2018.05.010

Modi, K., & Dayma, R. (2017). Computing and control (I2C2), review on fraud detection methods in credit card transactions. *2017 International Conference on Intelligent, IEEE*. https://doi.org/10.1109/I2C2.2017.8321781

Olszewski, D. (2014). Fraud detection using self-organizing map visualizing the user profiles. *Knowledge-Based Systems*, *70*, 324–334. https://doi.org/10.1016/j.knosys.2014.07.008

Pandey, Y. (2017). Credit card fraud detection using deep learning. *International Journal of Advanced Research in Computer Science*, *8*(5), 981–984.

Patidar, R., & Sharma, L. (2011). Credit card fraud detection using neural network. *International Journal of Soft Computing and Engineering (IJSCE)*, *1*, 32–38.

Patil, S., Nemade, V., & Soni, P. K. (2018). Predictive modelling for credit card fraud detection using data analytics. *Procedia Computer Science*, *132*, 385–395. https://doi.org/10.1016/j.procs.2018.05.199

Pumsirirat, A., & 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*(1), 18–25. https://doi.org/10.14569/IJACSA.2018.090103

Roy, A. (2018). Deep learning detecting fraud in credit card transactions. *Stems and Information Engineering Design Symposium (SIEDS), 2018 Sy, IEEE*. https://doi.org/10.1109/SIEDS.2018.8374722

Ruder, S. (2016). An overview of gradient descent optimization algorithms. arXiv preprint arXiv:1609.04747.

Sadgali, I., Sael, N., & Benabbou, F. (2019). Performance of machine learning techniques in the detection of financial frauds. *Procedia Computer Science*, *148*, 45–54. https://doi.org/10.1016/j.procs.2019.01.007

Sahin, Y., & Duman, E. (2011). Detecting credit card fraud by ANN and logistic regression. *2011 International Symposium on Innovations in Intelligent Systems and Applications, IEEE*.

Sasirekha, M. (2012). *An integrated intrusion detection system for credit card fraud detection. Advances in computing and information technology* (pp. 55–60). Springer.

Sharma, A. & Panigrahi, P. K. (2013). A review of financial accounting fraud detection based on data mining techniques. *International Journal of Computer Application,* 39(1), 37–47.

Sorournejad, S., Zojaji, Z., Atani, R. E., & Monadjemi, A. H. (2016). A survey of credit card fraud detection techniques: Data and technique oriented perspective. arXiv preprint arXiv:1611.06439.

Sureshkumar, B. (2019). Survey of fraud spotting techniques. *Journal of the Gujarat Research Society*, *21*(17), 89–94.

Tripathi, K. K., & Pavaskar, M. A. (2012). Survey on credit card fraud detection methods. *International Journal of Emerging Technology and Advanced Engineering*, *2*(11), 721–726.

West, J., & Bhattacharya, M. (2016). Intelligent financial fraud detection: a comprehensive review. *Computers & Security*, *57*, 47–66. https://doi.org/10.1016/j.cose.2015.09.005

Yomas, J., & Kiran, C. N. (2018). Critical analysis on the evolution in the e-payment system, security risk, threats and vulnerability. *Communications on Applied Electronics*, *7*(23), 21–29. https://doi.org/10.5120/cae2018652800

Zareapoor, M., Seeja, K. R., & Afshar Alam, M. (2012). Analysis on credit card fraud detection techniques: based on certain design criteria. *International Journal of Computer Applications*, *52*(3), 35–42. https://doi.org/10.5120/8184-1538

Zhang, Y., & Trubey, P. (2019). Machine learning and sampling scheme: An empirical study of money laundering detection. *Computational Economics*, *54*(3), 1043–1063. https://doi.org/10.1007/s10614-018-9864-z