

Credit Card Fraud Identification Using Machine Learning

Andre Juliantama

18217040

Sistem dan Teknologi Informasi

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung

Bandung, Indonesia

juliantamaandre@gmail.com

Abstract— Frauds are known as an act of deception that is committed by a person or more to get an unlawful gain or profit for themselves. Frauds generally are dynamic, and they have no patterns. Thus, identifying a fraud is a challenge itself. Fraudsters usually take advantage of recent technological developments. In one way or another, they are able to bypass security checks and find the flaw of the system to cheat the flow of transactions which leads to a lot of losses. However, fraudulent activities can be prevented by machine learning techniques. This study will cover Convolutional Neural Network (CNN), Long Short-term Memory (LSTM), Hybrid of CNN-LSTM, K-Nearest Neighbor (KNN), Random Forest and Support Vector Machines (SVM). The datasets that will used are from European, Australian and German datasets and from Indonesian local bank.

Keywords-*fraud, identification, credit card, machine learning, CNN, LSTM, CNN-LSTM, KNN, Random Forest, SVM*

I. Introduction

Credit cards are now generally used by almost everyone as a method of payment for their transactions, be it online transactions or offline transactions. However, using credit cards

may expose us to several threat. Fraudulent transaction is one of them. Fraudulent transaction is done by the fraudsters to make unauthorized purchases. They somehow found a way to exploit the credit card owners then steal their identity to make those transactions. Many efforts have been made to reduce and prevent fraudulent activities. Despite those efforts, fraudulent activities are still out there happening. Opportunities to make fraudulent transactions are always following the technological advancement.

Identifying fraudulent transaction is using computer system is a challenge itself. The problem is that the data available is imbalanced. For the past several years, studies that tried to identify fraudulent transaction resulted in a plenty of methods to develop a strong classifier that maximize classification accuracy and minimize two results: (1) false positive which upset customers and merchants, and (2) false negative which destroy wise image of bank [1].

Normally, identifying fraudulent transaction is divided into two categories [2]. *First*, supervised approach: the models are trained to learn patterns using given labeled samples of fraudulent and non-fraudulent transaction. *Second*, unsupervised approach: the models are trained to detect abnormal transactions from training dataset. The assumption of the second approach is that abnormal transaction might be possible fraudulent transaction case. Stated by these approaches, the objective of identifying fraudulent transaction it to predict the probability of a transaction labeled as fraudulent transaction.

Machine learning usage has been proved successful to solve classification problems in many areas thus encourage researchers to use these models to identify fraudulent transactions. Many learning models have been proposed such as: Random forest, SVM, KNN, LSTM, CNN. These researches mostly preceded by preliminary studies to extract features from past historical transaction data that represent the pattern of purchasing habit or transaction of the customers. Then the transaction is represented by a set of features as input to the selected machine learning models.

The problems present in identifying fraudulent transactions are mainly: (1) there are no feature standard of how the transaction is represented; (2) the imbalanced data distribution between fraudulent transaction and non-fraudulent transaction. The number of fraudulent transactions is less than the number of non-fraudulent transactions; (3) lack of dataset to validate models issued by previous studies due to banking confidential reasons; and (4) less

classification of fraudulent and non-fraudulent transactions as credit card fraudsters usually replicate their target consumer behavior as similar as the non-fraudulent transactions.

In this paper, we will be looking into 4 data sets. They are the European dataset, the Australian dataset, the German dataset and Indonesian local bank dataset. The work would aim to benchmark and compare different machine learning models. Best performing models would be presented in the conclusions by comparing the results on the datasets, except the Indonesian local bank. Indonesian local bank dataset will be approached using different machine learning models.

The sections of this paper are arranged as follows. Section 2 summarizes the related work and the background of the models. Section 3 will explain methods and analysis. Section 4 will present and evaluate the results. Finally, conclusions will be summarized in section 5.

II. Related Work

A. Comparative study on KNN and SVM

A paper on Fraud Detection using different techniques namely Naïve Bayes, KNN, SVM and Bagging Ensemble Classifier was published by Zareapoor and Shamsolmoali. In their paper, they discuss the various concerns while handling this problem, such as there is a non-availability of real-world data which forces researches to work on simulated data as the banks and other financial institutions don't make their data public due to privacy concerns as it is a sensitive data. Most of the available data are highly unbalanced as the number of fraudulent transactions is only 2% and 98% of the transactions are legal. They refer to the concerns of big data and the computation time it takes for larger datasets in these cases. One of the major challenges which frequently reported in many research papers is about the dynamic nature of fraud. The nature of fraud cannot be defined by one particular situation or style. Thus, it needs machine learning algorithms to be updated regularly so that malicious attempts could be caught in real time.

In their experiment, they used the data from the UCSDFICO competition where it contained 100,000 instances of credit card transaction with 20 attributes from an ecommerce website. Only 2293 instance/transactions were 2019 International Conference on Computational Intelligence and Knowledge Economy

(ICCIKE) December 11–12, 2019, Amity University Dubai, UAE 334 fraudulent which makes the ratio of legitimate to fraud as 100:3.

The methodology they used was splitting the datasets into 4 different sets which contained 20% fraud, 15% fraud, 10% fraud and 3% fraud. In their experiment, they realized that the accuracy or error rate would not be the best evaluation measure in this case. Hence, they used True Positive Rate (fraud catching rate), True Negative Rate, False Positive Rate (false alarm rate) and False Negative rate. These four values help in reflecting the performance more than the accuracy and the error rate. They used a 10-cross-fold validation for their experiments. The final result showed that KNN had a much better false alarm rate and fraud catching rate as compared to SVM and Naïve Bayes Classifier on all four sub-datasets [3].

B. Random Forest in Fraud Detection

Randhawa et al's paper on Credit card fraud detection using AdaBoost and majority voting explores many different machine learning algorithms such as Naïve Bayes, Random Forest, Gradient Boosted Tree, etc. In this paper, they use "Majority Voting" for combining two or more algorithms. The study also investigates AdaBoost ensemble model and reports that AdaBoost is very sensitive to anomalies and outliers.

They use the RapidMiner as an implementation Software and the experiments are conducted using the South-East Asia region credit card data. This dataset is a highly imbalanced data set with less than 1% fraud transaction. All of the classifiers were evaluated using 10-fold cross validation to reduce the bias. The classifier performance is evaluated using the Matthews Correlation Coefficient (MCC).

MCC helps in measuring the efficiency of the 2-class problem by taking TP, TN, FP and FN rates into account. In their experiment, they found that Random Forest had one of the best MCC rates at 0.990 compared to other methods such as SVM, gradient boosted trees, etc. By using AdaBoost, Random Forest with achieved 100% accuracy and MCC rate to be 1. In this study the generalization

ability of the developed model should be carefully examined as it is not clear how good it would perform on unknown data.

In general, the study reports that hybrid approaches produce more reliable results compared to single classifier models [3].

C. Convolutional Neural Networks (CNN)

CNN is a variant of feed-forward neural network and is a powerful model to address classification problem. CNN has been successfully used to address classification problems in many research fields for the past 20 years. In General, CNN is made of two main parts: feature extraction part and classification part. The feature extraction includes one or several convolution layers and pooling layers. The extracted output (feature map) are then used to be the input to the classification part.

CNN capability to capture local dependencies of features in the input data is its strength. This model has triggered many researchers to proposed variants of CNN that is tested using various image datasets such as MNIS, traffic sign, CIFAR10, Imagenet, PASCAL VOC.

Given a training dataset $\{x_n\}$ for $n = 1, 2, \dots, N$ and corresponding set of labels as target $\{t_n\}$. Neural network is trained by minimizing the Mean Square Error (MSE) or E function formulated as follows.

$$E(w) = \frac{1}{2} \sum_{n=1}^N \|y(x_n, w) - t_n\|^2 \quad (1)$$

Figure 1. E Function

Where: $y(x_n, w)$ is the predicted label as output of the CNN given the vector input x_n and the vector weight w resulted from the training algorithm and t_n the corresponding actual label.

The study by Hinton, Srivastave, Krizhevsky, Sutskever, and Salakhutdinov concluded that several factors have become enabler to the advancement of CNN model performance were mainly: (1) the availability of large labeled training datasets; (2) high computation power supported by GPU server; and (3) model regularization. The success of CNN performance in image classification has

concerned many researchers to adopt CNN models for fraud transaction detection [1].

D. Long Short-term Memory (LSTM)

Long Short-term Memory (LSTM) model is a variant of recurrent neural network (RNN). LSTM can be viewed as a network of LSTM units (see Figure 2). Each LSTM unit is equipped with three gates that control information flow: (1) input gates to decide when the input is significant enough to remember; (2) forget gate to decide when the unit should continue to remember or forget the value; and (3) output gate to decide when the unit should output the value.

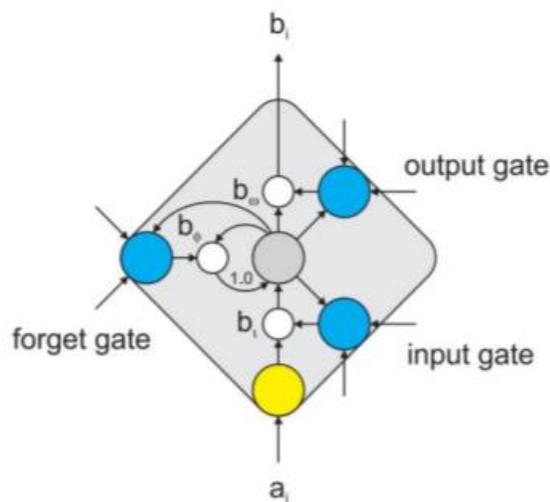


Figure 2. LSTM unit (Source: [1])

In the past decade, LSTM models have been recognized as powerful models which show learning capability from sequence data. The strength of LSTM lies in its ability to capture long range dependencies and learn effectively from varying length sequences. Several studies have reported that LSTMs performed well to solve the following problems: framewise phoneme classification, sound classification, scene image classification, video classification, image generation, and medical diagnostic. LSTM models also have been explored to recognize fraudulent card transaction [1].

E. Hybrid of CNN and LSTM (CNN-LSTM)

Many studies in the past five years have showed that the combined CNN and LSTM model produced a more robust model than CNN and LSTM respectively. The CNN-LSTM model works so that input data is first processed by CNN. The

CNN output is then processed by LSTM as classifier. High performance of CNN-LSTM model is due combining capability of CNN to captured short-term feature relations and LSTM to capture longer-term temporal feature relations. For example, the study by in video classification; in 86 sentiment classification; in video-based emotion recognition; in phrase-aware sentence classification; and in abnormal event detection from video [1].

III. Methods and Analysis

A. Datasets

i. European, Australian, German

The European Dataset contains the transactions made by credit card users in 2 days in September 2013. All of the fields except time and amount have been PCA transformed. It contains only 492 fraud instances out of 284,807 instances.

Both the Australian Dataset and the German Dataset are acquired from the UCI ML repository. The data sets were anonymized such that no personal information is provided.

The Australian contains 383 normal instances and 307 fraud instances. In the same time the German data set contains 1000 instances; of which 700 of them are normal instances and 300 of them are fraud instances. The European dataset is considerably larger than the Australian and the German datasets [3].

This evaluation will try to conclude which machine learning models is most suitable to identify credit card fraud.

ii. Indonesian Local Bank

The data provided is a historical debit card transaction dataset in the period of 2016-2017 provided for this research by a local Indonesia bank under permission. The transaction data has been labeled fraudulent or non-fraudulent transactions by the card issuer bank. Due to imbalance between nonfraud and fraud samples, imbalanced data treatment is mainly under-

sampling technique to reduce the number of samples from majority class using under-sampling ratio R such that $1 \leq R \leq 4$. That is, nonfraud dataset is resampled to produce 4 datasets with the following composition (See Table 1).

Table 1. Under-Sampling Input Dataset

Dataset	R	N_{Fraud}	N_{NonFraud}
1	1	2,896 (50%)	2,896 (50%)
2	2	2,896 (33%)	5,792 (67%)
3	3	2,896 (25%)	8,688 (75%)
4	4	2,896 (20%)	11,582 (80%)

The under-sampling ratio (R) is defined as: $R = \frac{N_{NonFraud}}{N_{Fraud}}$

Where: $N_{NonFraud}$ is the number of non-fraudulent transaction samples and N_{Fraud} is the number of fraudulent transaction samples [1].

B. Experimental Setup (European, Australian, German Dataset)

All the implementations were conducted in Python and using various libraries such as NumPy, Pandas, Keras, Scikit-Learn, and Tensorflow. Rstudio was occasionally used for data cleaning. For K-nearest neighbor, cross-validation is used on the training data set to determine the best value of the neighbors K for each data set. The best K for each data set is then used to conduct further analysis on the entire data set. For Support vector machines and Random Forest, a grid-based search approach is used to find the best parameter for each model. Python method GridSearchCV is used.

With CNN, convert the dataset into a 2D array instead of a 1D array. The data passes through a series of convolutional layers and max-pooling layers followed by a layer to flatten the data. Finally, the data is classified at SoftMax layer.

The models then tested using cross-validation and choose the **top three** performing models and combine them using majoring voting. The diagram below show the basic structure of the model [3].

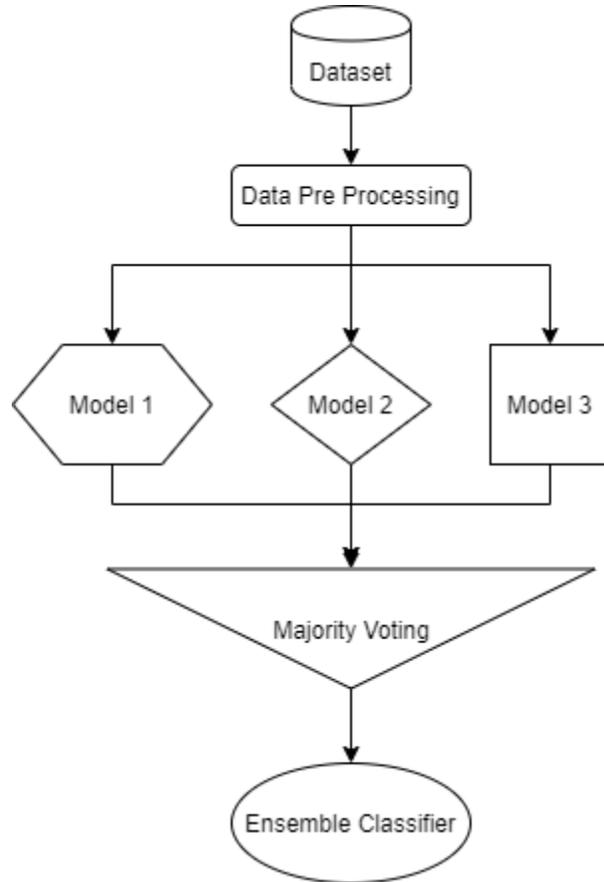


Figure 3. Majority Voting Based Model Structure

C. Financial Transaction Feature (Indonesian Local Bank Dataset)

To capture short-term and long-term relationship of spending pattern in using the cards, the financial transaction features in this study are built based on the following principles.

- i. This represents financial transaction by a sequence of transaction amounts within a given period. The feature is represented by a vector that captures historical transaction amount in the past 31 days. Consider x be historical transaction unit of a card-ID where:

$$x = (x_1, x_2, x_3, \dots, x_{50})$$

Where $x_1, x_2, x_3, \dots, x_{50} \in \mathcal{R}$. Each of these features are defined as follows:

- x_1, x_2, \dots, x_{28} : daily transaction amount from the past 28 days period.
- $x_{29}, x_{30}, \dots, x_{48}$: average transaction amount for each 2 consecutive days in the past 1 month.
- x_{49} : is the minimum transaction amount in the past 31 days period; and x_{50} : the maximum transaction amount in the past 31 days period.

In the transaction representation, zero paddings are added if the number of calendar days of a month is less than 31. To re-scale the transaction amount, each transaction unit then transformed using log function as follows.

$$x_t = \log(x + 1)$$

Where: 1 is a vector of 1's.

Let X be transaction unit dataset, Y be transaction unit labels, and m be the number of transaction units. The dataset and the corresponding labels can be represented as follows.

$$X^T = [x_{1,t}, x_{2,t}, \dots, x_{m,t}]$$

$$Y^T = [y_1, y_2, \dots, y_m]$$

- ii. Dimensional reduction using Principal Component Analysis (PCA): PCA is an orthogonal transformation to reduce the number of features (dimension) of input dataset which is computed as follows [1].

$$X_t = X P$$

Where: X_t the transformed dataset; $P = [v_1, v_2, \dots, v_k]$ is $k \times k$ matrix whose columns are the eigenvectors of $X^T X$, $k \leq 50$ is the number of Principal Components of $X^T X$; and $Cov_X = X^T X$ is covariance matrix of the input data X .

D. Deep Learning Models (Indonesian Local Bank Dataset)

The deep learning models which were explored in this study are: CNN, Stacked LSTM, and CNN-LSTM. The structures of each model are illustrated for length of input data = 30 and batch size = 20 (See table 2, 3, 4) [1].

Table 2. THE STRUCTURE OF CNN-LSTM (INPUT LENGTH = 30)

Layer (Type)	Output Shape	N-Parameters
Embedding	(None, 30, 30)	3,000
Conv1D	(None, 30, 20)	1,820
MaxPooling	(None, 10, 20)	0
Conv1D	(None, 10, 20)	1,220
MaxPooling	(None, 3, 20)	0
Dropout	(None, 3, 20)	0
Batch normalization	(None, 3, 20)	80
LSTM	(None, 100)	48,400
Dropout	(None, 100)	0
Batch normalization	(None, 100)	400
Dense	(None, 1)	101

Table 3. THE STRUCTURE OF CNN (INPUT LENGTH = 30)

Layer (Type)	Output Shape	N-Parameters
Embedding	(None, 30, 30)	3,000
Conv1D	(None, 30, 20)	1,820
MaxPooling	(None, 15, 20)	0
Conv1D	(None, 15, 20)	1,220
MaxPooling	(None, 7, 20)	0
Dropout	(None, 7, 20)	0
Dense	(None, 1)	141

Table 4. THE STRUCTURE OF SLSTM (INPUT LENGTH = 30)

Layer (Type)	Output Shape	N-Parameters
Embedding	(None, 30, 30)	3,000
Dropout	(None, 30, 30)	0
Batch normalization	(None, 30, 30)	120
LSTM	(None, 30, 30)	7,320
Dropout	(None, 30, 30)	0
Batch normalization	(None, 30, 30)	120
LSTM	(None, 30, 30)	7,320
Dropout	(None, 30, 30)	0
Batch normalization	(None, 30, 30)	120
LSTM	(None,30)	7,320
Dropout	(None,30)	0
Batch normalization	(None,30)	120
Dense	(None, 1)	31

E. Evaluation Criteria

i. European, Australian, German Dataset

The Matthews Correlation Coefficient is a measure to evaluate the quality of a two class/binary classifier. It was proposed by Brain W. Matthews in 1975. The coefficient returns +1 for a perfect prediction, while a value zero indicates a random prediction. Matthews Correlation is also known as the phi coefficient. Davide Chicco mentions that MCC is a much better measure than accuracy and F1 score as the other two can be misleading because they do not consider all four values of the confusion matrix.

ROC curve is the receiver operation characteristic. It helps in determining the precision of the model because of the imbalance in the dataset. ROC curve is basically plotting of TPR on the x axis against FPR on the y axis. Sometimes two ROC curves may have the similar Area under the curve (AUC), then in that case, we need to look further into the finer details such as the Cost of failure.

The idea behind the cost of failure is that each of the False Negatives (Frauds detected as Normal) would have a cost of \$1000 and False Positives

(Normal instances detected as fraud) would have a cost of \$100 to the company/entity. We are using this method to evaluate the top three models because sometimes all three of them have very similar MCC and AUC values. Similarly, the cost of the resulting ensemble classifiers is also calculated [3].

ii. Indonesian Local Bank Dataset

Several metrics widely used to evaluate classifier performance in classifier tasks are as follows. x Accuracy:

1. Accuracy is a common classifier performance metric to measure the degree of closeness of predicted class label using the trained classifier to the actual class label. Formula to compute accuracy is as follows:

$$Accuracy = \frac{TF+TN}{N}$$

Where: TF is true positive, TN is true negative, and N is the number of datasets. Therefore, the higher the accuracy value, the higher classifier performance that is represented.

2. Receiver Operating Characteristic (ROC) Curve. ROC curve is a two-dimensional curve that plots the false positive rate as horizontal axis against the true positive rate as vertical axis. Each pair value of the true positive rate (TPR) and the false positive rate (FPR) at various threshold settings is computed and then plotted in the ROC curve. Formula to compute each pair of (TPR, FPR) value is as follows:

$$\left(\frac{FP}{FP+TN}, \frac{TP}{TP+FN} \right)$$

Where: FP is false positive, FN is false negative, TP is true positive, TN is true negative,

3. Area under the ROC Curve (AUC): is a scalar value whose value between 0 and 1 that represents expected performance of classifier. The statistical property of AUC is equivalent to the probability that the classifier will rank a randomly selected positive sample higher than a randomly selected negative sample. Therefore, the higher the AUC value, the higher classifier performance that is represented [1].

IV. Results and Evaluations

A. European, Australian, German Dataset

The tables list the Matthew Correlation Coefficient (MCC), the Area Under the Curve measure (AUC) and cost of failure for a set of learning models. The methods are ordered base on their value of MCC.

i. European Dataset

Table 5. European Dataset Results

Method	MCC	AUC	Cost of Failure
Random Forest	0.7947	0.8507	30340
CNN	0.8096	0.8764	25700
SVM	0.8145	0.9004	21220
KNN	0.8354	0.8887	22660
Ensemble (KNN, SVM and CNN)	0.8226	0.8964	21740

Table 5 show the result from the European Dataset. Random Forest shows good result of MCC and AUC but has the highest cost. CNN, SVM, and KNN have the best performance in terms of MCC and AUC. SVM has the least cost of failure. The majority voting classifier is constructed from the top three performing models. The ensemble method performs better than SVM and CNN individually, but it has a similar cost to SVM. However, SVM has better AUC value. So, the recommendation for the European dataset will be SVM if the company look for the method that has least cost as much as possible. The ensemble method would take longer time in terms of both training and testing while SVM has the least in terms of testing and training.

ii. Australian Dataset

Table 6. Australian Dataset Results

Method	MCC	AUC	Cost of Failure
CNN	0.6408	0.8227	6430
Random Forest	0.684	0.8416	4700
KNN	0.6905	0.8425	6460
SVM	0.7085	0.8551	3380
Ensemble (KNN, SVM, Random Forest)	0.7281	0.8655	3470

Table 6 summarizes the result from the Australian Dataset. SVM, KNN and Random Forest have the best performance. Ensemble method achieve higher score in MCC and AUC compared to SVM individually with a little bit higher cost of failure. Thus, the Ensemble method will be chosen due to its best MCC and AUC value and considerable cost of failure.

iii. German Dataset

Table 7. German Dataset Results

Method	MCC	AUC	Cost of Failure
KNN	0.2725	0.5873	21100
Random Forest	0.2912	0.6437	16970
SVM	0.4038	0.6857	16400
CNN	0.4291	0.7056	14220
Ensemble (SVM, CNN, Random Forest)	0.4439	0.7011	15620

Table 7 shows the result from German Dataset. Random Forest, SVM and CNN show the best result in terms of performance in MCC and AUC values. Their cost of failure value is also better. An ensemble of SVM, CNN and Random Forest is built.

iv. Overall Results

In general, the results in tables 5, 6, 7, for the three datasets show that combining the top performing models outperform single models. Ensemble improvement is clearer on smaller dataset (the German and Australian dataset). For European dataset, the ensemble produced a result a little less than SVM.

Random Forest works best for smaller number of datasets. CNN was found to be the best learning method as it produces good results for both European dan German dataset, while its performance was on the 4th best in the Australian dataset. Its cost of failure is similar to KNN. For the German dataset, it has the lowest cost as well.

Table 8 summarizes the frequency of individual models ranking at the top 3 performing models for all datasets. SVM consistently was the best performing models of all datasets. KNN also produces good results with both large dataset and the smaller datasets [3].

Table 8. Top Performing Models

Method	Number of times in Top 3
Support Vector Machines	3 Times
K-Nearest Neighbors	2 Times
Convolutional Neural Networks	2 Times
Random Forest	2 Times

B. Indonesian Local Bank Dataset

Distribution and cumulative of Eigenvalues from the input data covariance matrix is shown in the Figure 4 and Figure 5. Eigenvalues from the first 20, 30 and 40 Principal Components contributes to 68.7 %, 87.0 %, and 95.1 % of cumulative Eigenvalues respectively [1].

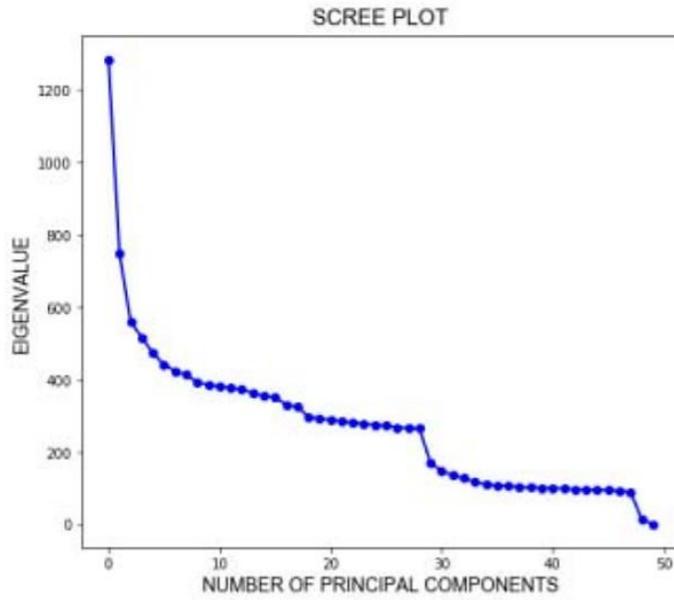


Figure 4. Scree Plot of Eigenvalues of Covariance Matrix

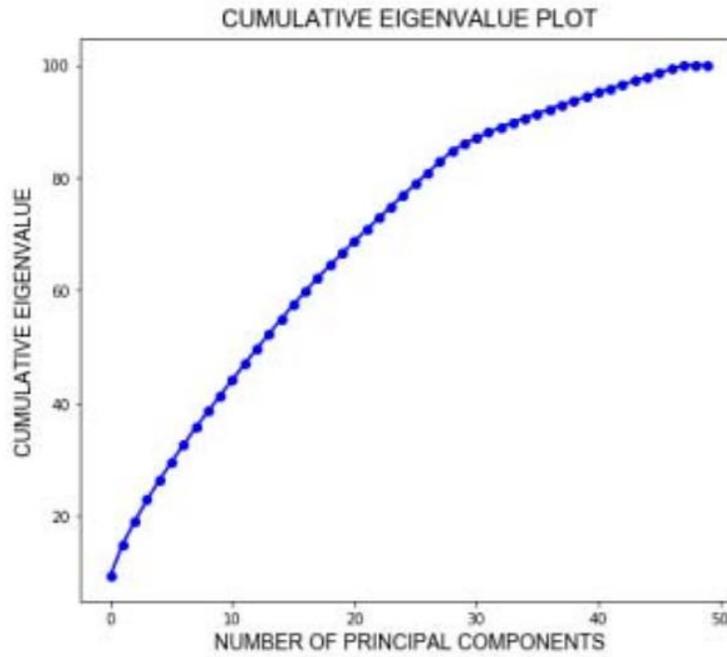


Figure 5. Cumulative Eigenvalues of Covariance Matrix

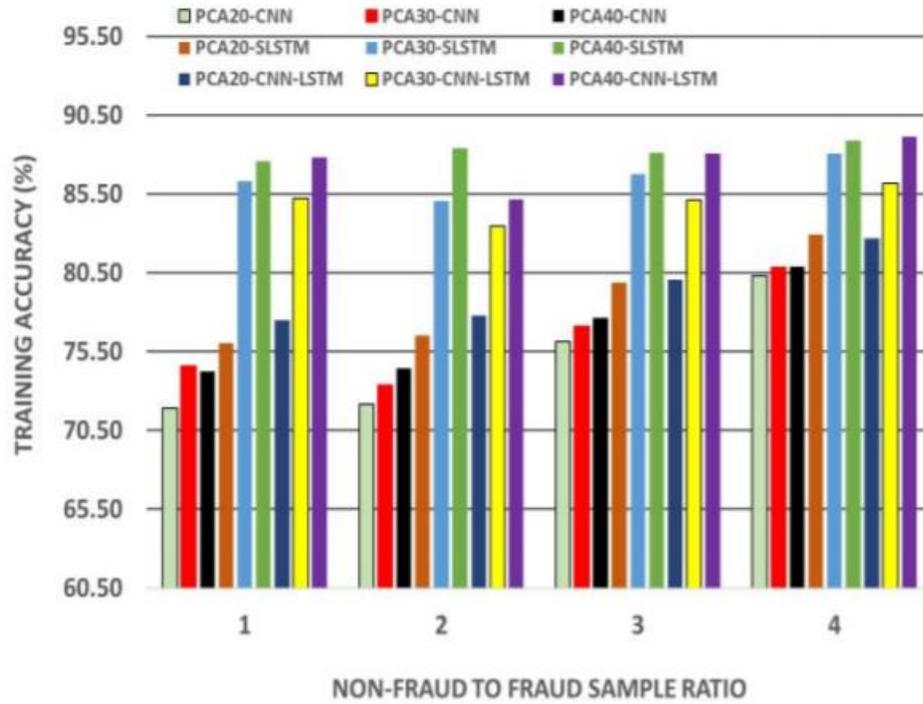


Figure 6. Training Accuracy of CNN, Stacked LSTM, and CNN-LSTM

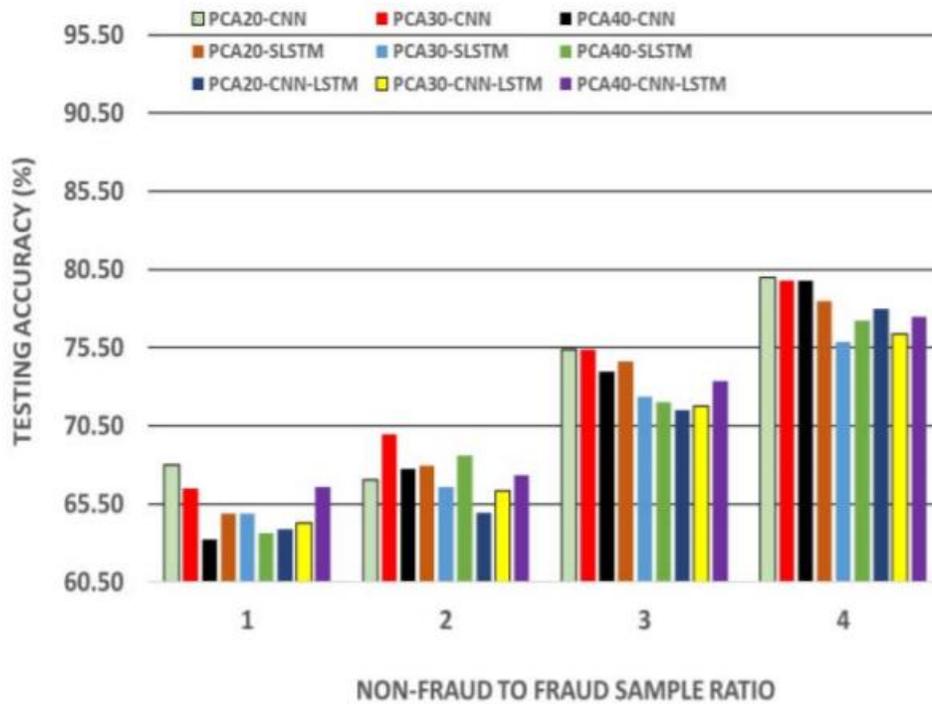


Figure 7. Testing Accuracy of CNN, Stacked LSTM, and CNN-LSTM

Based on these results, the raw input dataset was transformed using Eq. (8) with $k = 20, 30, 40$ to produce 3 transformed datasets. Each of the transformed dataset was used to train, validate, and test models supervised with epochs = 500 and Adam optimizer algorithm. Model cross-validation used leave one-out technique. The results of supervised training and testing of each model can be summarized in the Fig. 4 and 5. AUC from each model is shown in Fig. 8. There are three main findings [1]:

1. As expected, the higher nonfraud to fraud ratio the higher training and testing accuracy. However, due to imbalanced data between nonfraud and fraud, accuracy cannot be used solely as model performance indicator.
2. For each nonfraud to fraud ratio, SLSTM achieve the highest training accuracy followed by CNN-LSTM and CNN models.
3. AUC metric from testing results showed that CNN performance achieved the best results followed by CNNLSTM and SLSTM. CNN performance models are more stable than performance of the other models. These results indicate that the fraudulent transaction patterns are dominated by short term transaction amount relationship rather than long-term ones.

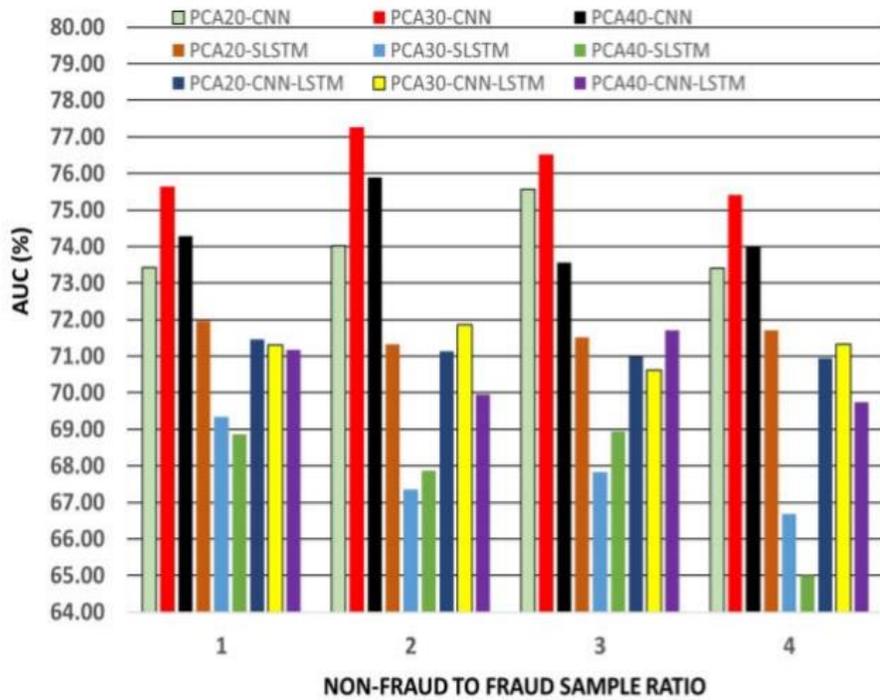


Figure 8. Area Under the Curve (AUC) of Receiver Operating Characteristic

V. Conclusion

Research about fraud identification has been around for longer than we thought, and many methods and techniques have been used to identify fraudulent transactions. Be it manual checking or autonomous checking. Machine learning models have been proven to succeed in this domain. Many considerations have been taken to choose which learning models best fit the condition based on different available factors. Based on the two studies, various models of machine learning method have been used to predict and identify possible fraudulent transaction. On both studies, CNN shows great performance for recognizing fraudulent data. However, this only limit to supervised learning. Before choosing which model best fit to our need, first and foremost we must consider several factors that might affect the impact from the result of machine learning models. Number of data, gain and loss analysis and cost of failure are several example of factors that should be considered.

References

- [1] Y. Heryadi and H. L. H. S. Warnars, "Learning temporal representation of transaction amount for fraudulent transaction recognition using CNN, Stacked LSTM, and CNN-LSTM," *2017 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom)*, Phuket, 2017, pp. 84-89.
- [2] N. Carnerio, G. Figueira, and M. Costa. "A data mining based system for credit-card fraud detection in e-tail," *Decision Support Systems*. 95(1), pp. 91-101, 2017. doi: 10.1016/j.dss.2017.01.002
- [3] P. Raghavan and N. E. Gayar, "Fraud Detection using Machine Learning and Deep Learning," *2019 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)*, Dubai, United Arab Emirates, 2019, pp. 334-339.
- [4] P. Kumar and F. Iqbal, "Credit Card Fraud Identification Using Machine Learning Approaches," *2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT)*, CHENNAI, India, 2019, pp. 1-4.