# Credit Card Fraud Detection using Deep Learning based on Auto-Encoder and Restricted Boltzmann Machine

Apapan Pumsirirat, Liu Yan
School of Software Engineering, Tongji University
Shanghai, China

**Abstract—Frauds have no constant patterns. They always change their behavior; so, we need to use an unsupervised learning. Fraudsters learn about new technology that allows them to execute frauds through online transactions. Fraudsters assume the regular behavior of consumers, and fraud patterns change fast. So, fraud detection systems need to detect online transactions by using unsupervised learning, because some fraudsters commit frauds once through online mediums and then switch to other techniques. This paper aims to 1) focus on fraud cases that cannot be detected based on previous history or supervised learning, 2) create a model of deep Auto-encoder and restricted Boltzmann machine (RBM) that can reconstruct normal transactions to find anomalies from normal patterns. The proposed deep learning based on auto-encoder (AE) is an unsupervised learning algorithm that applies backpropagation by setting the inputs equal to the outputs. The RBM has two layers, the input layer (visible) and hidden layer. In this research, we use the Tensorflow library from Google to implement AE, RBM, and H2O by using deep learning. The results show the mean squared error, root mean squared error, and area under curve.**

***Keywords—Credit card; fraud detection; deep learning; unsupervised learning; auto-encoder; restricted Boltzmann machine; Tensorflow***

## I. INTRODUCTION

Fraud detection in online shopping systems is the hottest topic nowadays. Fraud investigators, banking systems, and electronic payment systems such as PayPal must have an efficient and complex fraud detection system to prevent fraud activities that change rapidly. According to a CyberSource report from 2017, the present fraud loss by order channel, that is, the percentage of fraud loss in their web store was 74 percent and 49 percent in their mobile channels [1]. Based on this information, the lesson is to determine anomalies across patterns of fraud behavior that have undergone change relative to the past.

A good fraud detection system should be able to identify the fraud transaction accurately and should make the detection possible in real- time transactions. Fraud detection can be divided into two groups: anomaly detection and misuse detection [2]. Anomaly detection systems bring normal transaction to be trained and use techniques to determine novel frauds. Conversely, a misuse fraud detection system uses the labeled transaction as normal or fraud transaction to be trained in the database history. So, this misuse detection system entails a system of supervised learning and anomaly detection system a system of unsupervised learning. What is the difference between supervised learning and unsupervised learning? The answer is that supervised learning studies labeled datasets. They use labeled datasets to train and to render it accurate by changing the parameters of the learning rate. After that, they apply parameters of learning rate to the dataset, the techniques that implement supervised learning such as multilayer-perceptron (MLP) to build the model based on the history of the database. This supervised learning has a disadvantage, since if new fraud transactions happen that do not match with the records of the database, then this transaction will be considered genuine. While, unsupervised learning acquires information from new transactions and finds anomalous patterns from new transaction. This unsupervised learning is more difficult than supervised learning, because we have to use appropriate techniques to detect anomalous behavior.

Neural networks were introduced to detect credit card frauds in the past. Now, we focus on deep learning that is a subfield of machine learning (ML). Based on deep learning in the first period, they use deep learning to know about an image's processing. For example, Facebook uses deep learning in the function to tag people and to know who the person is for subsequent reference. Further, deep learning in neural networks have many algorithms for use in fraud detection, but in this paper, we selected the AE and RBM to detect whether normal transaction of datasets qualified as novel frauds. We believe that some normal transaction in datasets that were labeled as fraud also show suspicious transaction behavior. So, in this paper we focus on unsupervised learning.

In this paper, we use three datasets in these experiments; these datasets are the German, Australian, and European datasets [4], [3], [18]. The first dataset is German, provided by Professor Dr. Hans Hofman [4]. There are twenty attributes that describe the capability, such as credit history, purpose to use credit card, credit amount, job, among others. The German dataset were 1000 instances. The second dataset is from Australia. [3] The attributes' names and values in this dataset have been changed to meaningless symbols to protect the confidentiality of the data. There were 690 instances. The last dataset was from a European cardholder from September 2013. This dataset shows the transaction that occurred in two days with 284, 807 transactions. There were 31 features in this dataset. The 28 features, such as V1, V28 is a numerical input variable result of PCA transformation. Other 3 feature that do

not bind with PCA are "Time", "Amount", and "Class". This experiment will bring together three datasets to compare different receiver operating characteristics (ROC) to understand the performance of binary classifiers.

## II. Related Work

In the past decade, credit card was introduced in the financial segment. Now, credit card has become a popular payment method in online shopping for goods and services. Since the introduction of credit cards, fraudsters have tried to falsely adopt normal behavior of users to make their own payments. Due to these problems, most research on credit card fraud detection has focused on pattern matching in which abnormal patterns are identified as distinct from normal transactions. Many techniques for credit card fraud detection have been presented in the last few years. We will briefly review some of those techniques below.

The K-nearest neighbor (KNN) algorithms are used to detect credit card frauds. This technique is a supervised learning technique. KNN is used for classification of credit card fraud detection by calculating its nearest point. If the new transaction is coming and the point is near the fraudulent transaction, KNN identifies this transaction as a fraud [5]. Many people confuse KNN with K-means clustering, whether they are the same techniques or not. K-means and KNN are different. K-means is an unsupervised learning technique, used for clustering. K-Means tries to determine new patterns from the data and by clustering the data into groups. Conversely, KNN is the number used to compare the nearest neighbor to classify or predict a new transaction based on previous history. The distance in KNN between two data instances can be calculated by using different method, but mostly by using the Euclidean distance. KNN is very useful.

The outlier detection is another method used to detect both supervised and unsupervised learning. Supervised outlier detection method studies and classifies the outlier using the training dataset. Conversely, unsupervised outlier detection is similar to clustering data into multiple groups based on their attributes. N. Malini and Dr. M. Pushpa mention that the outlier detection method based on unsupervised learning is preferred to detect credit card fraud over outlier supervised learning, because unsupervised learning outlier does not require prior information to label data as fraudulent. So, it needs to be trained by using normal transactions to discriminate between a legal or illegal transaction [5].

Some credit card fraud transaction datasets contain the problem of imbalance in datasets. Anusorn Charleonnan mentions that the unbalance of datasets has many characteristics that emerge during the classification. He uses RUS, a data sampling technique, by trying to relieve the problem of class unbalance by editing the class distribution of training datasets. There are two major methods of adjusting the imbalance in datasets, undersampling and oversampling. In his research, he also uses the MRN algorithm for the classification problem of credit card fraud [6].

Artificial neural network (ANN) is a flexible computing framework used to solve a comprehensive range of non-linear problems. The main idea of ANN is mimicking the learning algorithm of the human brain. The smallest unit of ANN is called a perceptron, is represented as a node. Several perceptrons are connected as a network like the human brain. Each node has a weighed communication with several other nodes in the adjacent layer. A weight is simply a floating-point number, and it can be adjusted when the input eventually comes to train the network. Inputs are passed from input nodes through hidden layers to output nodes. Each node can learn and adjust itself to make it more accurate and appropriate.

The problem of credit card fraud detection has been analyzed with the Chebyshev Function Link Artificial Neural Network (CFANN). CFANN consists of two components, functional expansion and learning. Mukesh Kumar Mishra and Rajashree Dash, authors who used CFANN to detect credit card fraud by comparing it with MLP, and the Decision Tree [7]. MLP infers that the topology was structured into a number of layers. The first layer is called input layer, the middle layer is called the hidden layer. This layer can have more than one layer, and the last layer is called the output layer. Feed forward infers that all information flows in the same direction, the left-to-right direction, without recurrent links. Decision Tree is a structured tree that has a root node and a number of internal and leaf nodes. Their paper compares the performance of CFANN, MLP, and Decision Tree. The result of their study suggests that MLP outperforms CFANN and Decision Tree in fraud detection. Conversely, CFANN makes accurate predictions over the other two techniques [7].

Deep learning forms a state of the art technology in the present day. Most people in IT should follow this. First, ANN was introduced. After that, ML becomes a subset of ANN, and deep learning, a subfield of ML. Deep learning has been used in many fields such as image recognition in Facebook, speech recognition in Apple or Siri, and natural language processing in Google translator. Yamini Pandey used deep learning with the H2O algorithm framework to know complex patterns in the dataset. H2O is an open source for predictive data analytics on Big Data. Supervised learning is based on predictive analytics. The author used H2O based multi-layered, feed forward neural network to find credit card fraud patterns. H2O's performance based on the deep learning model shows less error in mean squared error, root mean squared error, mean absolute error, and root mean squared log error. Hence, these errors are low that enhances accuracy. The model's accuracy is also high in relation to the errors mentioned above [8]. Another concern before registering credit cards is credit cards' analysis' judgement. Ayahiko Niimi uses deep learning to judge whether a credit card should be issued to the user if they satisfy particular criteria. Transaction judgement refers to the validity of a transaction's attributes before making the decisions. To verify the transaction, the author uses the benchmark experiment based on deep learning and confirms that the result of deep learning has similar accuracy as the Gaussian kernel SVM. For the comparison, the authors use five typical algorithms and change the parameters of deep learning for five times, such as activation function and dropout parameter [9].

## III. Deep Learning Technique for Detect Credit Card Fraud

Deep learning is the state of the art technology that recently attracted the IT circle's considerable attention. The deep learning principle is an ANN that has many hidden layers. Conversely, non-deep learning feed forward neural networks have only a single hidden layer. The given picture shows the comparison between non-deep learning as in Fig. 1 and deep learning with hidden layers as in Fig. 2.

Now, we know about ANN, ML, and Deep Learning (DL). If these three words are metaphorically equated with the human body, they would be comparable as follows: artificial intelligence is like the body that contains the traits of intelligence, reasoning, communication, emotions, and feeling. ML is like one system that acts in the body, especially the visual system. Finally, deep learning is comparable to the visual signaling mechanism. It consists of a number of cells, such as retina that acts as a receptor and translates light signals into nerve signals. Now, we shall compare all the three categories with the human body.

Deep learning is a generic term used for multilayer neural network. Based on deep learning, there are many algorithms to implement such as AE, deep convolutional network, support vector machine, and others. One problem in selecting the algorithm to solve the problem is that the developer should know the real problem and what each algorithm in deep learning does. The three algorithms of deep learning that do unsupervised learning are RBM, AE, and the sparse coding model. Unsupervised learning automatically extracts the meaningful features of your data, leverages the availability of unlabeled data, and adds a data-dependent regularization for training.

In this study, we use AE for credit card fraud detection. AE has the input equal to the output in the hidden layer that has more or less the kind of input units depicted in the Fig. 3.

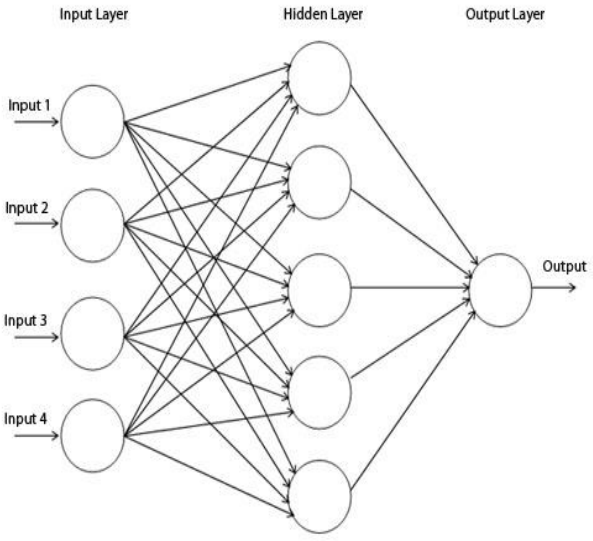


Fig. 1. Single layer hidden neural network [10].

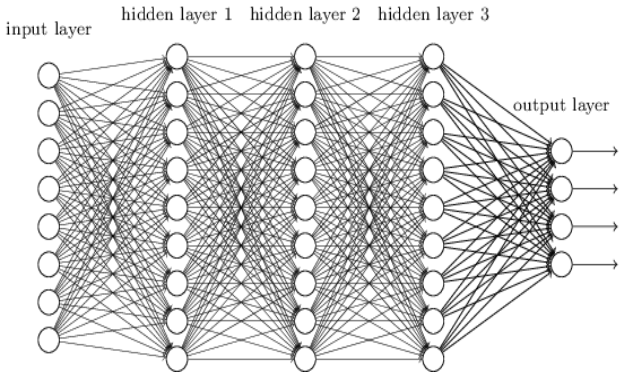


Fig. 2. Deep neural network [11].

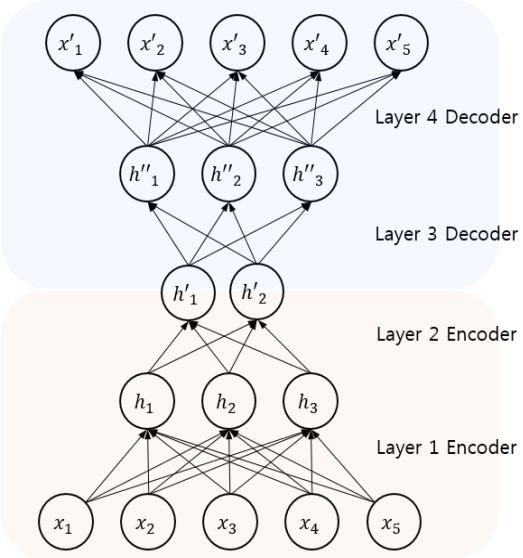


Fig. 3. Auto-encoder [12].

The equation of an encoder and a decoder are presented here:

| | | |
|---|---|---|
| *Encoder* | | |
| | $h(x)$ | $= g(a(x))$<br>$= sigm(Wx)$ or<br>$= \tanh(Wx)$ |
| *Decoder* | | |
| | $\hat{x}$ | $= o(\hat{a}(x))$<br>$= sigm(W * h(x))$ or<br>$= \tanh(W * h(x))$ |

In this study to implement AE, we use the hyperbolic tangent function or "tanh" function to encode and decode the input to the output. As a sample of a neural network, when we have already used the AE model, we should reconstruct the error by using backpropagation. Backpropagation computes the "error signal", propagates the error backwards through network that starts at the output units by using the condition that the error forms the difference between the actual and desired output values. Based on the AE, we use parameter gradients for realizing backpropagation.

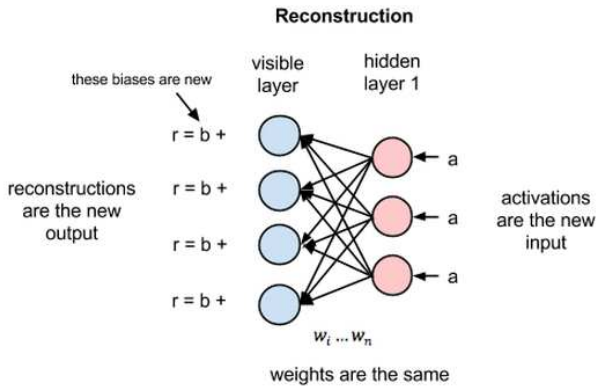


Fig. 4. Reconstruction of RBM [17].

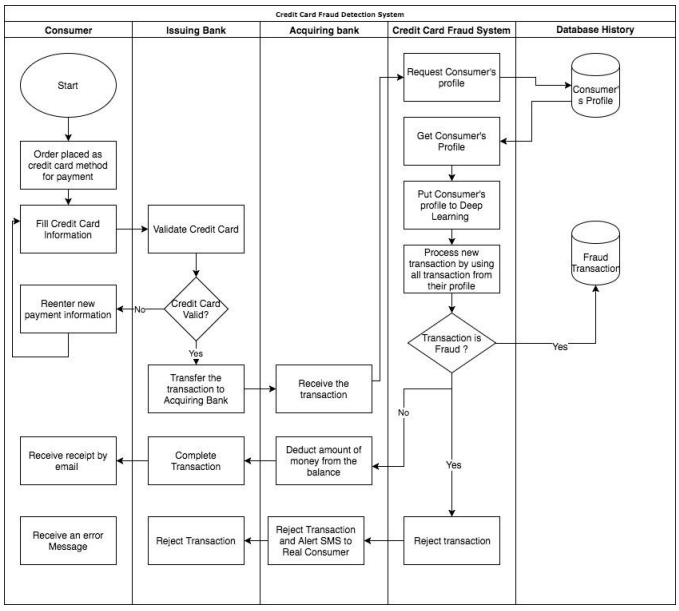


Fig. 5. Credit card fraud detection system using deep leaning.

Another algorithm is RBM. There are two structures in this algorithm, visible or input layer and hidden layer. Each input node takes the input feature from the dataset to be learned. The design is different from other deep learning, because there is no output layer. The output of RBM is getting the reconstruction back to the input as shown in the picture below or Fig. 4. The point of RBM is the way in which they learn by themselves for data reconstruction; this is unsupervised learning.

Let us proceed to our design of credit card fraud detection system by using deep learning between AE and RBM in Fig. 5. First, the consumer orders the product via internet by using the credit card payment method. After that, the issuing bank sends the transaction to the acquiring bank by sending the amount of money, date and time of payment, location of internet usage, and more. Now, this is the credit card fraud detection system used to validate the behavior of credit card. As you can see, the credit card fraud system requests consumer's profile from the database to bring their behavior into the AE and RBM by using deep learning. Based on the AE, the acquiring bank transfers the input that is the amount of money, date and time, location of internet use, and other information. Then, the AE uses past behavior to be trained first, and then uses the new coming transaction as a validation test for the transaction. AE does not use labeled transactions to be trained, because it is unsupervised learning. RBM uses all transactions that transfer from acquiring bank as visible input and then that goes to the hidden node, and after the calculation of the activation function, the RBM reconstructs the model by transferring the new input from the activation function back to the output or visible function. As a conclusion of this in Fig. 5, if the transaction is fraudulent, the system will record this transaction as a fraud in the database and will then reject it. Next, the acquiring bank sends a SMS alert to the real consumer that the transaction has not been processed, because the system suspects the transaction as fraudulent.

## IV. Comparative Fraud Detection Techniques

Before focusing on the study of AE and RBM, this paper would prefer to compare it with other techniques to show that deep learning is suitable for finding anomalous patterns against normal transactions in Table I.

TABLE I. Comparison of Fraud Detection Technique

| **Fraud Detection Techniques** | **Advantage** | **Disadvantage** |
|---|---|---|
| K-nearest Neighbor Algorithm | KNN method can be used to determine anomalies in the target instance and is easy to implement. | KNN method is suitable for detecting frauds with the limitations of memory. |
| Hidden Markov Model (HMM) | HMM can detect the fraudulent activity at the time of the transaction. | HMM cannot detect fraud with a few transactions. |
| Neural Network | Neural networks have learned the previous behavior and can detect real-time credit card frauds. | Neural networks have many sub-techniques. So, if they pick-up this which is not suitable for credit card fraud detection, the performance of the method will decline. |
| Decision Tree | Decision Tree can handle non-linear credit card transaction as well. | Decision Tree have many type of input feature, DT can be constructed using different induction algorithm like ID3, C4.5 and CART. So, the cons are how to bring up induction algorithm to detect fraud as well. DT cannot detect fraud at the real time of transaction. |

| | | |
|---|---|---|
| Outlier Detection Method | Outlier detection detects the credit card fraud with lesser memory and computation requirements. This method works fast and well for large online datasets. | Outlier detection cannot find anomalies accurately like other methods. |
| Deep Learning | A key advantage of deep learning is the analysis and learning of a massive amount of unsupervised data. It can extract complex patterns [13]. | Now, deep learning is widely used in image recognition. No information to explain the other domains is available. The library of deep learning does not cover all algorithms. |

## V. Proposed Method

In this paper, we use Keras [15] as a high-level neural network API implemented by python. Another program that we implement in AE is H2O [16] package. We use the H2O package to find MSE, RMSE, and variable importance across each attribute of the datasets. Conversely, we used Keras in parallel processing to get AUC and confusion matrix. Both frameworks, we coded in python on Jupyterlab.

Before we could develop the program AE by using Keras API and code the program AE by using H2O, the datasets needed to be cleansed. As we know, the German credit card data set and the Australian dataset classified characteristics for each attribute. You can see the details of these attributes in [3], [4].

This is the step of cleansing data.

*1)* Classified the data into a number of classifications such as attribute 4 (qualitative) purpose

A40: Car (new)

A41: Car (used)

A410: others

We transform it to the number of classifications, such as A40 = 1, A41 =2, …, A410 = 10 and so on.

*2)* After obtaining the classification for each attribute, we transform those classifications into PCA by using XLSTAT [14].

TABLE II. Autoencoder Model using Keras

```
Input_Dimension = Training.shape[1]
Hiddenlayer = 16
Input_layer = Input(shape=Input_Dimension,))
Encoder1 = Dense(Hidden_layer,activation="tanh")(Input_layer)
Encoder2 = Dense(Hidden_Layer/2,activation="tanh")(Encoder1)
Encoder3 = Dense(Hidden_Layer/4,activation="tanh")(Encoder2)
Decoder1 = Dense(Hidden_Layer/4,activation="tanh")(Encoder3)
Decoder2 = Dense(Hidden_Layer/2,activation="tanh")(Decoder1)
Decoder3 = Dense(Input_Dimension,activation="tanh")(Decoder2)
AutoEncoderModel = Model(inputs=Input_layer,outputs=Decoder3)
```

TABLE III. Autoencoder Model using H2O

```
Autoencoder =
h2o.estimators.deeplearning.H2OAutoEncoderEstimator(hidden=hidden_structure, epochs=200, activation='tanh', autoencoder=True)
Autoencoder.train(x = Input,
                  Training_frame = data_set)
Print(Autoencoder)
```

In the Keras method, we designed 6 hidden layers by having 3 encoders and 3 decoders. In each hidden layer, we designed the following units:

| | |
|---|---|
| Input | : 21 attributes or 21 Input |
| Encoder1 (H1) | : 16 |
| Encoder2 (H2) | : 8 |
| Encoder3 (H3) | : 4 |
| Decoder3 (H1) | : 4 |
| Decoder2 (H2) | : 8 |
| Decoder3 (H3) | : 21 |
| Output | : 21 |

As mentioned above, every hidden layer we used was the "Tanh" activation function. In Keras, there are many activation functions to implement. Based on the experiment, we used "Tanh" function, because it achieves a high level of AUC. We divide the train and test with 80 and 20 percentage of data by using normal transactions to predict fraudulent transactions.

This is an example of Python Coding in Keras as in Table II.

As you can see, in Keras API, we need to build our model by preparing the command ourselves. Conversely, in the H2O package, we use the command of AE in Table III.

Base on methodology of our research, we coded in Python and then we used Area of Under Curve to identify the success rate of the model. If the percentage of AUC is high then mean that we found unsupervised learning rate with true positive rate on our model. Conversely, some datasets that has less amount of data will get more false positive rate because they has not much data to be trained.

## VI. Evaluate the Result

These are the result of the German Dataset show in Fig. 6, 7 and 8; as we mentioned above that the Dataset was divided for training and testing in a ratio 80:20 by using the normal labeled transactions in the column "Creditability" to find anomalous patterns. These form the AUC and confusion matrix.

This form the MSE and RSME from H2O the package of the German Dataset.

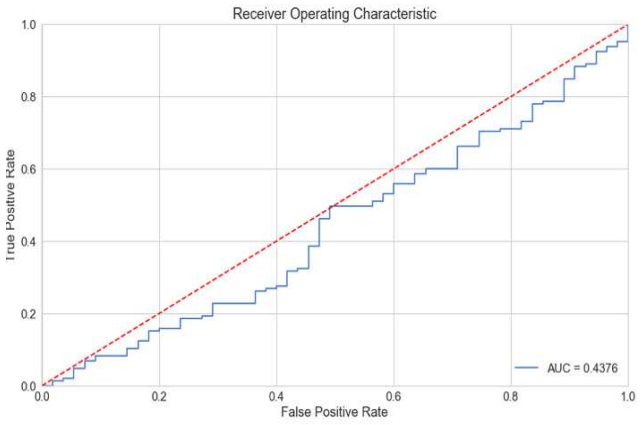


Fig. 6. AUC of German Dataset by using AE.

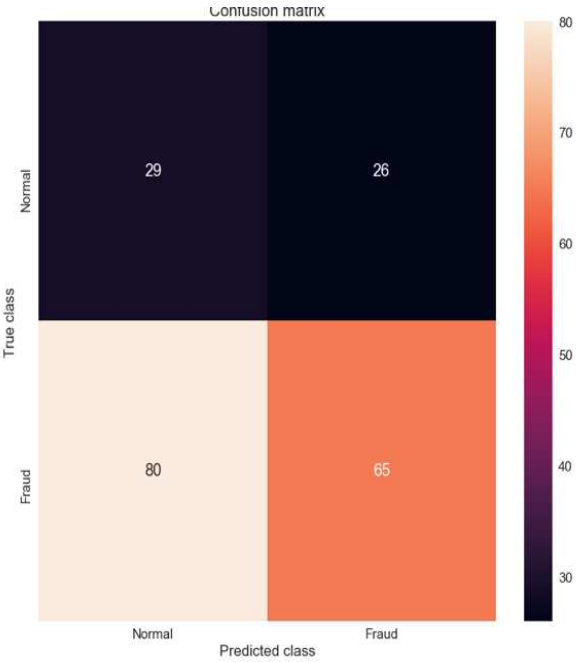


Fig. 7. Confusion Matrix of German Dataset by using AE.

```
Model Details
=============
H2OAutoEncoderEstimator :  Deep Learning
Model Key:  DeepLearning_model_python_1513057801244_1


ModelMetricsAutoEncoder: deeplearning
** Reported on train data. **

MSE: 0.00129114108292
RMSE: 0.0359324516687
```

Fig. 8. AE Model of deep learning report of German Dataset on H2O framework.

Let us move on to another dataset, the Australian Dataset. The AUC result is given, and the confusion matrix from Keras. The results are shown in Fig. 9, 10 and 11.

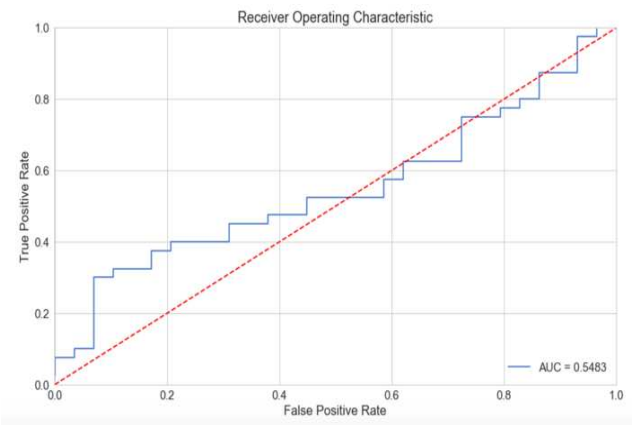


Fig. 9. AUC of Austrian Dataset by using AE.

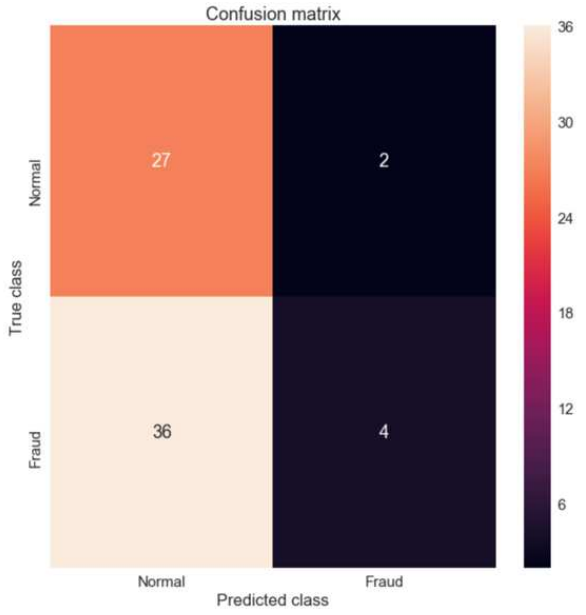


Fig. 10. Confusion Matrix of Australian Dataset by using AE.

```
Model Details
=============
H2OAutoEncoderEstimator :  Deep Learning
Model Key:  DeepLearning_model_python_1513081686672_1


ModelMetricsAutoEncoder: deeplearning
** Reported on train data. **

MSE: 0.000604421565799
RMSE: 0.0245849865934
```

Fig. 11. Auto Encoder Model deep learning report of Australian Dataset based on H2O framework.

This is the Australian Dataset's MSE and RSE obtained by running the H2O package.

Here, we move on to the large dataset, the European Dataset with 284, 807 transactions. The results are shown in Fig. 12, 13 and 14.

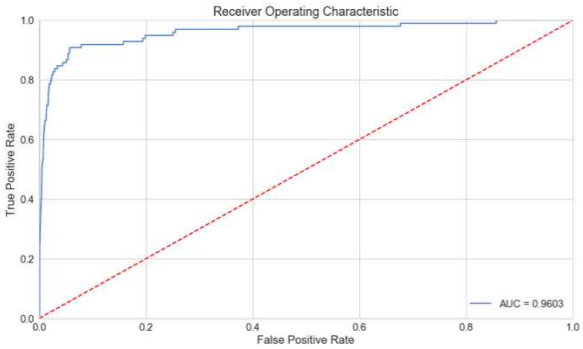


Fig. 12. AUC of European Dataset by using AE.

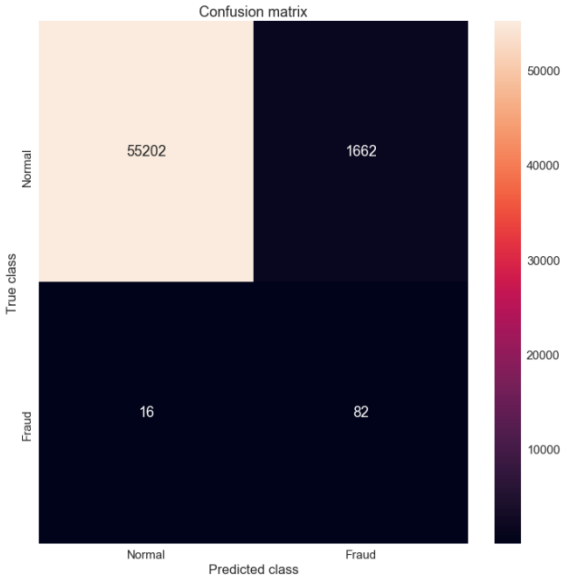


Fig. 13. Confusion Matrix of European Dataset by using AE.

```
Model Details
=============
H2OAutoEncoderEstimator :  Deep Learning
Model Key:  DeepLearning_model_python_1513687461083_1


ModelMetricsAutoEncoder: deeplearning
** Reported on train data. **

MSE: 1.30914842402e-05
RMSE: 0.0036182156155
```

Fig. 14. Auto Encoder Model deep learning report of European Dataset based on H2O framework.

As summarized by three datasets, there is lesser data in the German and Australian datasets. So, when we find anomalies in fraud detection, we obtain a lower of AUC, because we trained the systems for a small number of data and validated the test data for a lesser amount. Conversely, when we apply this AE model based on Keras with a large amount¸ the European Dataset, we got AUC of 0.9603. AE is suitable for large datasets.

Further RBM's results based on the three datasets are presented: we begin by explaining the German Dataset in Fig. 15. As you can see, the AUC of German Dataset is 0.4562.

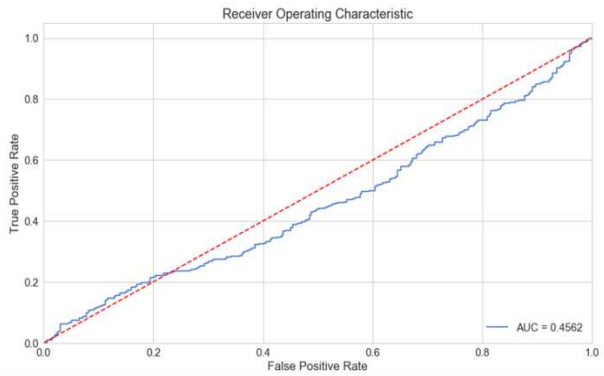


Fig. 15. AUC of German Dataset by using RBM.

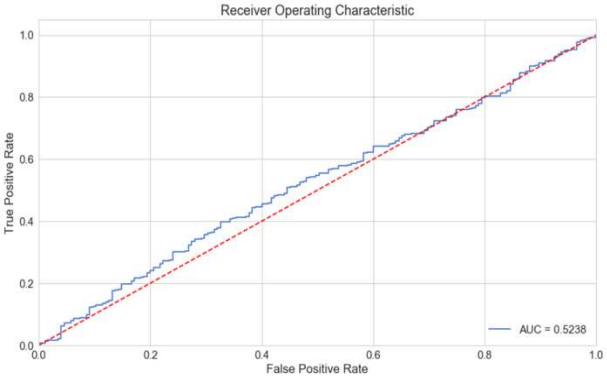


Fig. 16. AUC of Australian Dataset by using RBM.

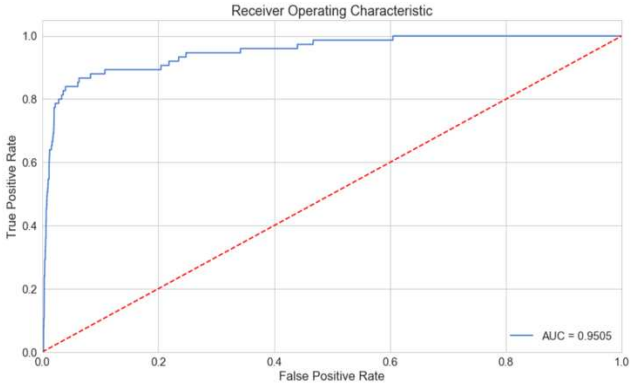


Fig. 17. AUC of European Dataset by using RBM.

The graph shows the result of the Australian Dataset by using the RBM algorithm to implement in Fig. 16. The AUC score is 0.5238.

While the biggest dataset is the European Dataset that produced an AUC value greater than the other two datasets shown above (Australian and German Dataset). The AUC score of European dataset is 0.9505 which can be seen in Fig. 17.

This is the summary of AUC's score that implemented AE and RBM of three different datasets.

From this research, we can conclude that AE and RBM produce high AUC score and accuracy for bigger datasets,

because there is a large amount of data to be trained. You can see the details of AUC's score in Table IV.

TABLE IV. COMPARISON AUC'S SCORE BETWEEN THREE DATASETS

| Dataset Name | No. of transactions | AUC's score based on AE | AUC's score based on RBM |
|---|---|---|---|
| **German Dataset** | 1000 | 0.4376 | 0.4562 |
| **Australian Dataset** | 690 | 0.5483 | 0.5238 |
| **European Dataset** | 284, 807 | **0.9603** | **0.9505** |

Based on two popular datasets, we can conclude that supervised learning dataset is suitable for history database for credit card fraud detection. Supervised learning such as multilayer perceptron in neural network that uses the prediction algorithm to identify whether new transactions are legal or illegal. When a credit card used, the neural network based on the fraud detection system checks for the pattern used by the fraudster and corroborates the pattern in question or checks for attributes that have been determined as illegal; if the pattern matches with genuine transaction behavior, then the transaction is considered legitimate. Conversely, unsupervised learning entails knowing about normal transactions and finding anomalous patterns, and then, responding in real-time to the system as a fraud or legal transaction.

## VII. CONCLUSION AND FUTURE WORK

Nowadays, in the global computing environment, online payments are important, because online payments use only the credential information from the credit card to fulfill an application and then deduct money. Due to this reason, it is important to find the best solution to detect the maximum number of frauds in online systems. AE and RBM are the two types of deep learning that use normal transactions to detect frauds in real-time. In this study, we focused on ways to build AE based on Keras, RBM, and H2O. To verify our proposed methods, we used benchmark experiments with other tools to confirm that AE and RBM in deep learning can accurately achieve credit card detection with a large dataset such as the European Dataset. Although, for these experiments, it will be better to use real credit card fraud transactions with a huge amount of data. We guarantee that AE and RBM can make more accurate AUC for receiver operator characteristics than that observable from the results from the European Dataset.

## REFERENCES

[1] CyberSource. (2017, Nov. 29). *2017 North AMERCA edition, online fraud benchmark report persistence is critical* [Online]. Available: http://www.cybersource.com/content/dam/cybersource/2017_Fraud_Benchmark_Report.pdf?utm_campaign=NA_17Q3_2017%20Fraud%20Report_Asset_1_All_Auto&utm_medium=email&utm_source=Eloqua

[2] L. Seyedhossein and M. R. Hashemi, "Mining information from credit card time series for timelier fraud detection," in 5th International Symposium on Telecommunications (IST'2010), 2010 © IEEE. doi: 978-1-4244-8185-9/10/$26.00

[3] UCI Machine Learning Repository. (2017, Nov. 29). *Stalog (Australian credit approval) dataset* [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Statlog+(Australian+Credit+Approval)

[4] UCI Machine Learning Repository. (2017, Nov. 29). *Stalog (German credit data) dataset* [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Statlog+%28German+Credit+Data%29

[5] N. Malini and Dr. M. Pushpa, "Analysis on credit card fraud identification techniques based on KNN and outlier detection" in 3rd International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEEICB17).

[6] A. Charleonnan, "Credit card fraud detection using RUS and MRN algorithm," in The 2016 Management and Innovation Technology International Conference (MITiCON-2016), 2016 © IEEE. doi: 978-1-5090-4105-3/16/$31.00

[7] M. K. Mishra and R. Dash, "A comparative study of chebyshev functional link artificial neural network, multi-layer perceptron and decision tree for credit card fraud detection" in 2014 13th International Conference on Information Technology, 2014 © IEEE. doi: 978-1-4799-8084-0/14 $31.00

[8] Y. Pandey, "Credit card fraud detection using deep learning" *Int. J. Adv. Res. Comput. Sci.*, vol. 8, no. 5, May–Jun. 2017.

[9] A. Niimi, "Deep learning for credit card data analysis," in World Congrss on Internet Security (WorldCIS-2015), 2015 © IEEE. doi: 978-1-908320-50/6 $31.00

[10] Single Hidden Layer Neural Network [Online]. Available: http://nicolamanzini.com/single-hidden-layer-neural-network/

[11] Chapter 6 (2017, Aug. 4). *Deep learning* [Online]. Available: http://neuralnetworksanddeeplearning.com/chap6.html

[12] Introduction Auto-encoder (2015, Dec. 21). *Auto-encoder* [Online]. Available: https://wikidocs.net/3413

[13] M. M. Najafabadi et al., "Deep learning applications and challenges in big data analytics," *J. Big Data*. doi: 10.1186/s40537-014-0007-7

[14] *XLSTAT your data analysis solution* [Online]. Available: https://www.xlstat.com/en/

[15] *Keras the python deep learning library* [Online]. Available: https://keras.io/

[16] *H2O API* [Online]. Available: http://docs.h2o.ai/h2o/latest-stable/h2o-docs/welcome.html

[17] *A beginner's tutorial for restricted Boltzmann machine* [Online]. Available: https://deeplearning4j.org/restrictedboltzmannmachine#define

[18] *Credit card fraud detection anonymized credit card transaction labeled as fruadulent or genuine* [Online]. Available: https://www.kaggle.com/dalpozz/creditcardfraud/data.