Journal of Computing and Applied Informatics (JoCAI) Vol. 6, No. 2, 2022 | 67 – 80

DATA SCIENCE
Journal of Computing and Applied Informatics

Predicting Fraudulence Transaction under Data Imbalance using Neural Network (Deep Learning)

Harry Patria

School of Computing, Newcastle University, England, United Kingdom

Abstract. The number of financial transactions has the potential to cause many violations of the law (fraud). Conventional machine learning has been widely used, including logistic regression, random forest, and gradient boosted. However, the machine learning can work as long as the dataset contains fraud. Many new financial technology companies need to anticipate the potential for fraud, which they have not experienced much. This potential for a crime can also be experienced by old service providers with a low frequency of previous fraud. With the data imbalance, traditional machine learningis likely to produce false negatives so that they do not accurately predict potential fraud. This study optimizes the machine learning approach based on Neural Networks to improve model accuracy through the integration of KNIME and Python Programming with KERAS and TensorFlow models. The study also conducts a comparative analysis to scrutinize the performance of Adam and Adamax Optimizer. Using data from European cardholders in 2013, this study proves that workflows and neural network algorithms can detect with up to 95% accuracy even with a very small fraud sample of only 0.17% or 492 of 284,807 transactions. In addition, the Adam optimizer performs higher accuracy than the Adamax optimizer. The implication is that this supervisory technology innovation can be developed to minimize transaction crimes in the financial services sector.

Keywords: Fraud, deep learning, neural networks, finance

Received 09 February 2022 | Revised 13 March 2022 | Accepted 05 April 2022

1 Introduction

Fraud is a classic transaction problem experienced by business owners, payment card service providers and payment service companies as well as the government. Although every stakeholder tries to suppress and prevent it, many have fallen victim to this crime. Nilson (2020) reported that fraud caused losses of up to 28.65 billion dollars, an increase of 2.9% from 27.85 billion dollars in 2018 (Nilson, 2020). The organization also projects that this crime could cost up to 40 billion dollars in 2027.

*Corresponding author at: School of Computing, Newcastle University, England, United Kingdom

E-mail address: harry.patria@sbm-itb.ac.id

ISSN: 2580-6769 | e-ISSN: 2580-829X | DOI: https://doi.org/10.32734/jocai.v6.i2-8309
Journal Homepage: https://jocai.usu.ac.id

Detection of crime in payment transactions is the most difficult challenge. With more and more data being generated from payment transactions, detection using human intelligence is becoming an impossibility. This has triggered the birth and development of many detection technologies using machine learning in the last decade (Borgne &Bontempi, 2021). On the other hand, the machine can detect quickly and accurately (Asha & Kumar, 2021; Zulfikri et al. 2021; Patria, 2021; Patria, 2022).

The machine learning works optimally by studying transaction features to distinguish between fraudulent and secure transactions. Various classification algorithms (supervised learning) are used to detect fraud. On the other hand, new service providers may not necessarily be able to detect crimes they have not experienced. With the limited number of transactions and the possibility of fraud at the beginning, conventional machine learning algorithms are not able to work optimally to detect fraud. Thus, many new service providers have the potential to experience various losses. The same thing also stalks old service providers with many ongoing transactions, the proportion of fraud against transactions will be much lower. So that the machine learning accuracy in detecting needs to be optimized (Zulfikri et al., 2021; Patria, 2021; Mariana & Patria, 2021).

The weakness of the fraud detection system is exploited by criminals through various cybercrimes including Skimming, Phishing, and Malware. Retrieval of customer data using a recording device in EDC machines and Automated Teller Machines (ATM) has often occurred until now (Skimming). In addition, the perpetrators also tricked the victim by creating a web similar to a banking transaction web by providing username and password features to steal customer data. With technological sophistication, perpetrators also use customer computers and web banking to penetrate internet banking transactions.

Credit cards have indeed become the most practical and efficient payment medium that is favored by the public to date. However, the high frequency and nominal use also trigger the potential for financial crimes to occur. Various companies use their experience to optimize service features to prevent various crimes that occurred previously, including by using machine learning.

The Financial Services Authority (OJK) through regulation Number 18/POJK.03/2016 seeks to improve the function of prevention and supervision through the implementation of risk management for banks. In supporting this function, supervisory technology has become a visionary breakthrough to implement the fraud eradication strategy. Thus, fraud can be minimized so that the public and banks can optimize financial transactions that have an impact on economic growth.

This research aims to design and optimize advanced machine learning for the detection of financial transaction crimes with data imbalance cases, which minority group is around 1% of 1

million samples (Johnson and Khoshgoftaar, 2019). Conventional machine learning can predict as long as there is sufficient fraud data. However, if it is too extreme, then the advanced approach using artificial neural networks (deep learning) is a breakthrough that is starting to be widely used by various global companies. Through this approach, the engine can detect with high accuracy using the integration of the KERAS and TensorFlow models and frameworks. Thus, machines can optimize the number of iterations of neurons and hidden layers of deep learning models that adopt the analogy of human neural networks. The implication is that this research can detect fraud accurately and quickly even though the proportion of fraud data compared to the number of transactions is not balanced.

2 Literature Review

Financial crime differs from industry to industry. In the financial sector, financial frauds can be divided into 4 (four) groups, namely banking, corporations, insurance, and cryptocurrencies found in Figure 1. Credit card fraud has become the most popular and has attracted the attention of academics and practitioners to apply machine learning (Al-Hashedi&Magalingam, 2021).

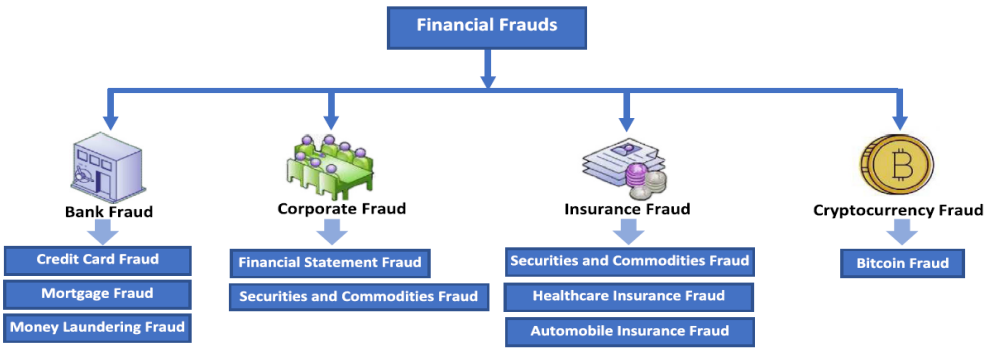


Figure 1. Fraud typology

Various academics and practitioners have researched the causes and methods of detecting fraud. Machine learning are starting to be widely used to detect anomalies that occur from big data that arises from the number of payment transactions. Al-Hashedi&Magalingam (2021) conducted a comprehensive review of data processing techniques for detecting fraud with machine learning models including Support Vector Machine (SVM), Fuzzy Logic (FL), Hidden Markov model (HMM), Artificial Neural Network (ANN), K-Nearest Neighbor (KNN), Decision Tree (DT), Logistic Regression (LR) and Outliers Detection. The advantages and disadvantages of this model can be found in Table 1. Neural networks have been widely used for detecting cases with more complex features with larger data. Al-Hashedi&Magalingam (2021) have evaluated those 7 techniques used to classify fraud using artificial neural networks in the financial sector.

Table 1. Fraud detection algorithms

No	Model	description	Pros.	Cons.
1	Support Vector Machine (SVM)	Linear separator (hyperplane)	Relatively efficient Non-linear using Kernel	Computational resources for big data and features
2	Naïve Bayes	Group prediction	Relatively efficient	Moderate accuracy
3	Random Forest (RF)	Decision trees	Relatively fast Handling data imbalance	Inability to predict outside the trained data
4	Neural Network	Multi-layer network	Relatively higher accuracy Handling big data	Computational resource demand Overfitting potential
5	Logistic Regression (LR)	Binary classification	Relatively simple and efficient	Low accuracy
6	K-Nearest Neighbor (KNN)	Classification and regression	Relatively simple and efficient	Relied on k value Relatively low accuracy
7	Outliers Detection	Outlier detection	Labelled data	Inefficient Produce false positive
8	Decision Tree (DT)	Classification and regression using decision tree splitting	Relatively simple and efficient	Computational resources
9	Hidden Markov Model (HMM)	Random process	Statistical algorithm	Relatively complex
10	K-Means Clustering	Clustering the databased on distance	Effective and efficient	Have to define number of clusters Sensitive towards outliers

Based on the evaluation of the machine learning literature found in Table 1, Artificial Neural Networks (ANN) with artificial neural networks or genetic algorithms produce relatively better accuracy and performance than conventional machine learning algorithms such as SVM and LR. Second, the ANN technique has begun to be widely used to detect cases of extreme or unbalanced data. We recommend further machine learning methods to minimize false negatives that may occur due to an unbalanced amount of data. Fraud frequency which is relatively small compared to total transaction frequency can cause misclassification because machines have not learned enough to recognize fraud patterns. The implication is that the machine detects that there is no fraud (negative) even though in reality there is fraud.

In statistics or econometrics, errors caused by imbalanced data indicate a type 1 (one) error. The error occurs when the initial hypothesis is rejected and the hypothesis should be true. Thus, ordinary machine learning is not optimal for detecting fraud cases with unbalanced data.

3 Methodology

3.1 Data Collection

Dataset sourced from credit card providers reported by Université Libre de Bruxelles (ULB) to provide big data and fraud detection and data architecture found in Figure 2. The dataset can be found in a link http://mlg.ulb.ac.be. The data describes data on European cardholders from 284,807 transactions for 2 (two) days in September 2013. The data is not balanced because the percentage of fraud is only 0.17% of all transactions.

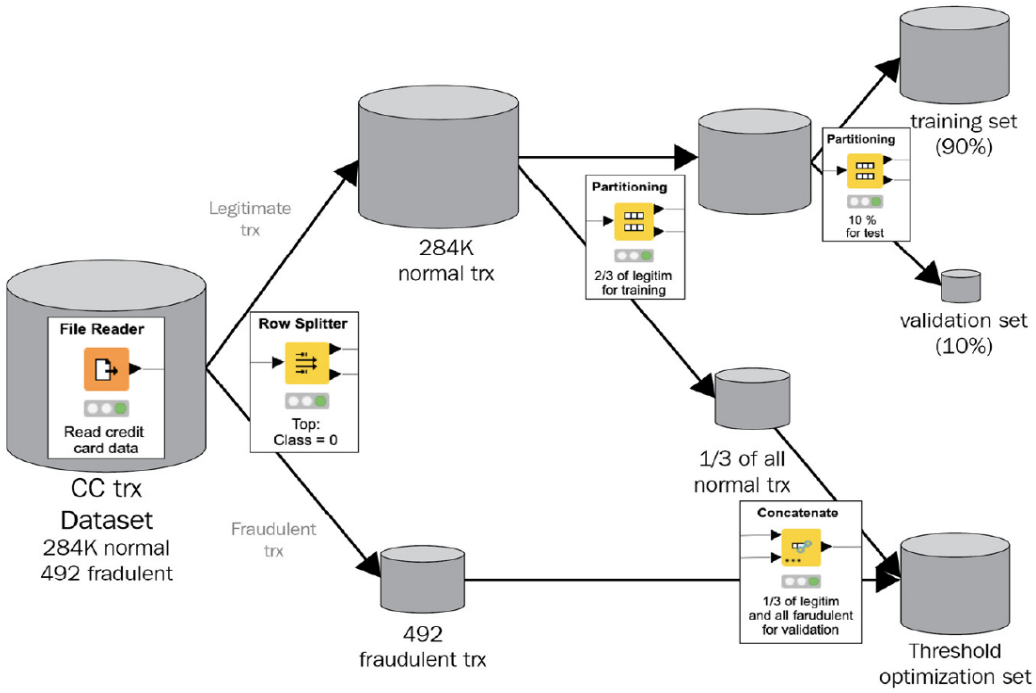
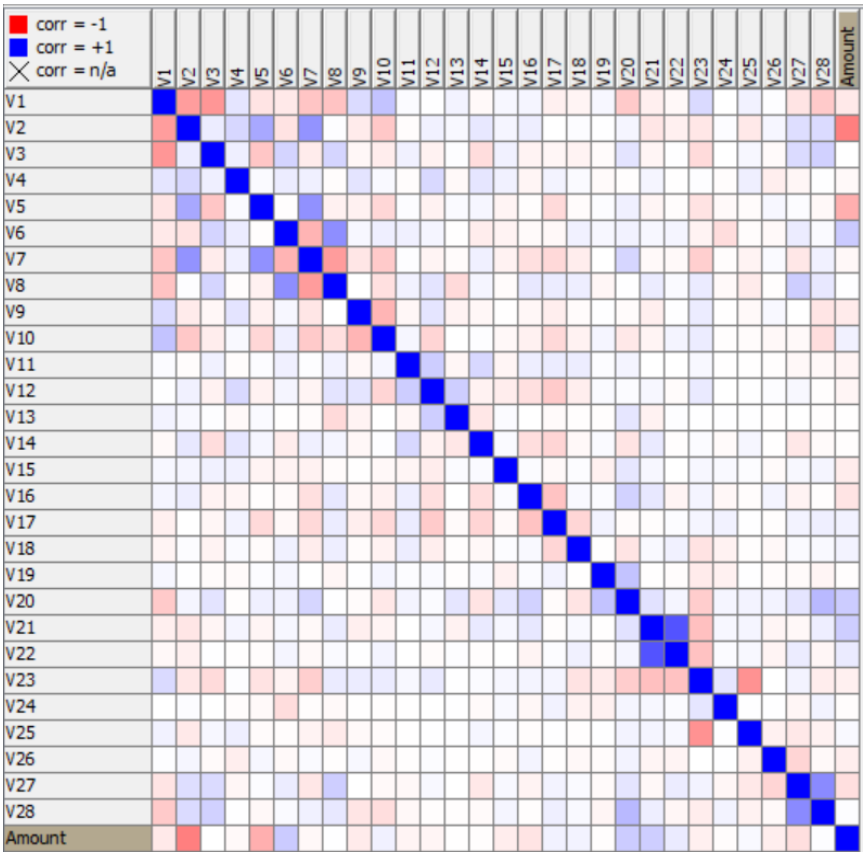


Figure 2. Data source and architecture

The data consists of 28 predictors or explanatory variables. The data already represents the key components of the results of data processing. To maintain the confidentiality of the data, European

cardholders released the data using Principal Component Analysis (PCA) for all data except for data on the time and nominal amount of transactions.

Figure 3. Heatmap correlation of data features



The dataset is processed by dividing the features by class without fraud (class = 0). Furthermore, the data is separated into 2 stages and with a random technique in node partitioning. As much as 10% of the data is used for validation which is then normalized. Meanwhile, the remaining 492 frauds of 284,807 transactions were used to validate with the rest of the first partition on the Concatenate node. The results are normalized first before being used to test the Keras model on the Keras Network Executor.

Regarding the Figure 3 resulted in KNIME tool, the correlation demonstrates the features used to predict a fraud label in this case. The result shows that both positive (blue) and negative (red) correlations among variables reflecting the features in the case of fraudulent transaction found in the heatmap plot.

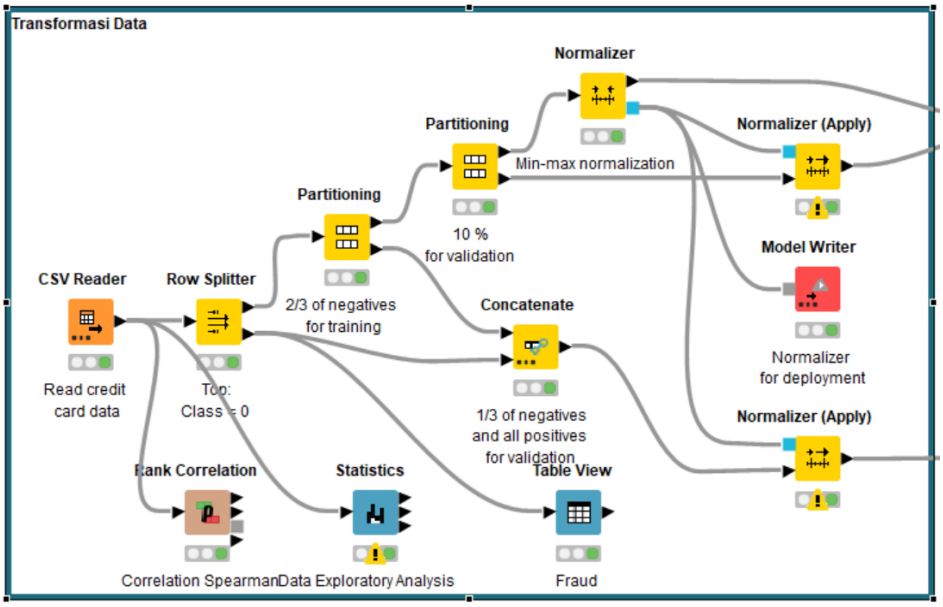


Figure 4. Pre-trained data processing and transformation

To prepare the data, we performed pre-trained data processing found in Figure 4. The first stage is to conduct descriptive analysis such as rank correlation and statistics. Later on, we transform data through splitter and partitioning to separate the dataset into a training and testing set. To deal with data unbalance, we perform sequential partitioning and concatenate to enrich the fraud and come up by normalizing the results.

3.2 Deep Learning

Deep learning (DL) is a set of algorithms that are part of artificial intelligence and machine learning found in Figure 5. Deep learning algorithms are tasked with training computational machines so that they can do work like humans in detecting objects, sound (audio), language translation, price prediction, fraud detection, and others.

A neural network is a collection of layers of nodes adopted like the human brain that contains neurons. This node is connected to other adjacent nodes. The more layers of nodes, the deeper the neural network is known as an artificial neural network. Each neuron in the human brain is interconnected and information flows from each neuron. Each neuron receives input and then operates with an amount in the form of a weighted sum and adds bias. These neurons are activated by an activation function that can be linear or non-linear.

The node will send an input signal and assign the appropriate weight. Nodes with heavier weight will have an impact on the next layer. The DL algorithm requires high hardware specifications to

process large amounts of data with complex mathematical computations. The DL algorithm works by developing many hidden layers.

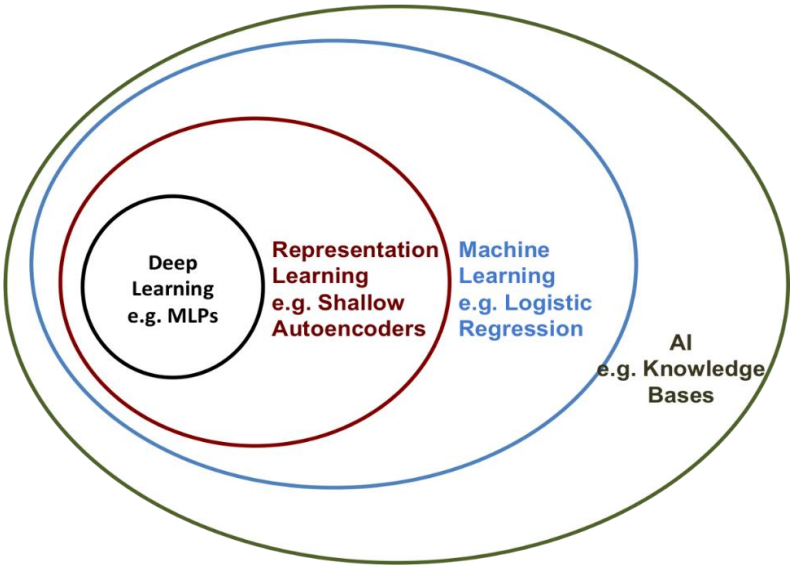
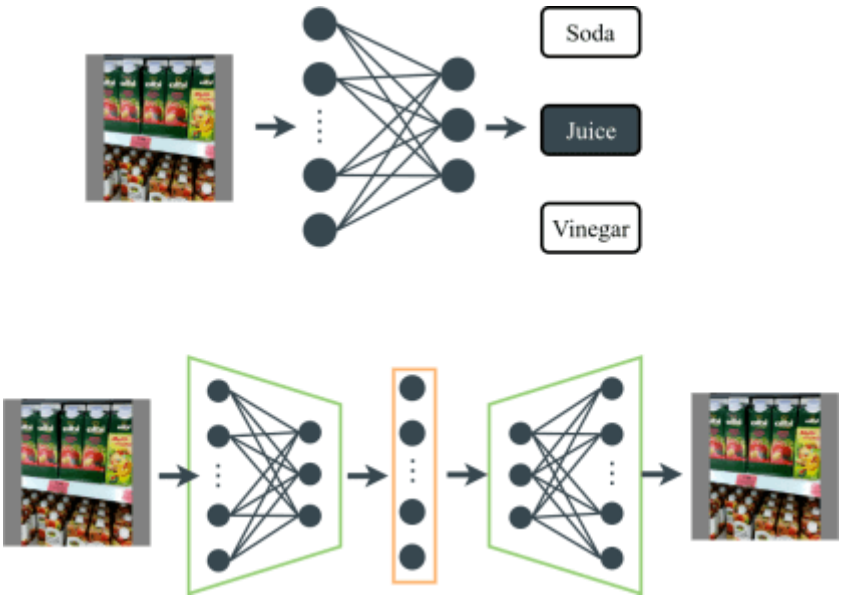


Figure 5. The structure of Artificial Intelligence and Deep Learning

DL variations include (i) Feedforward Neural Network; (ii) Recurrent Neural Network/Long Short-Term Memory (LSTM); and (iii) Convolutional Neural Network (CNN).Autoencoder is one of the artificial neural networks to "encode" data. In other words, the autoencoder is used to reduce the dimensions of the data without labels so that



it produces the same output as the input.

Figure 6. Traditional and Modern Neural Network using Encoder

The figure 6 demonstrates an architecture of ANN that is used to classify the appearance of foodstuffs. While the bottom is the autoencoder architecture. The middle layer of the

autoencoder is a vector collection of neurons (yellow box). This middle section usually has fewer neurons than the layers on the left and right (including the input and output layers). The implication is that a well-trained model can produce a simple representation of the input image. The vector is obtained from the input image through the architecture to encode. Furthermore, from this small vector, it can be regenerated (decoded) an image similar to the input.

3.3 Keras Autoencoder

This study develops Keras architecture with Autoencoder which consists of an input layer and an output layer and 5 layers of them (found in Figure 7) with sigmoid activation with a range from 0 to 1. The activation function is found in the Keras hidden (dense) layer and output layer, while the Keras input layer does not have an activation function.

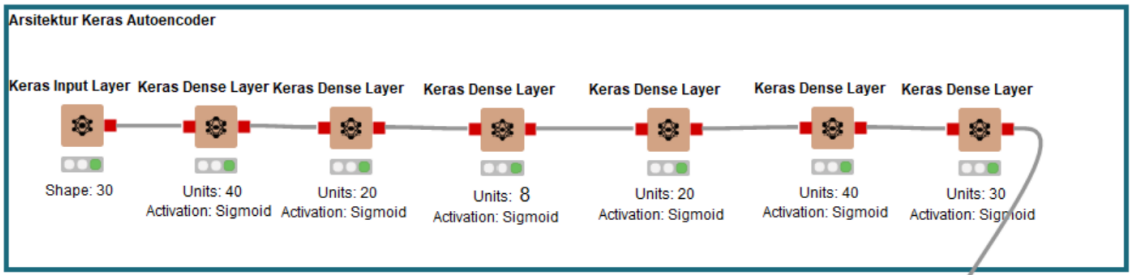


Figure 7. Keras Architecture using Encode

3.4 Adam Optimizer

Adam's algorithm is applied in this project which provides several advantages of the use of AdaGrad and RMSProp algorithms to optimize the algorithm for noisy cases. Second, Adam's algorithm is relatively easy to use by setting model parameters. On top of that, Adamax as a variant of Adam based on the infinity norm is used in this study to compare the results. According to the keras.io, Adamax is sometimes superior to Adam when it comes to models with embeddings. The model has a similarity with Adam while the epsilon is inserted for computational stability (Keras, 2022).

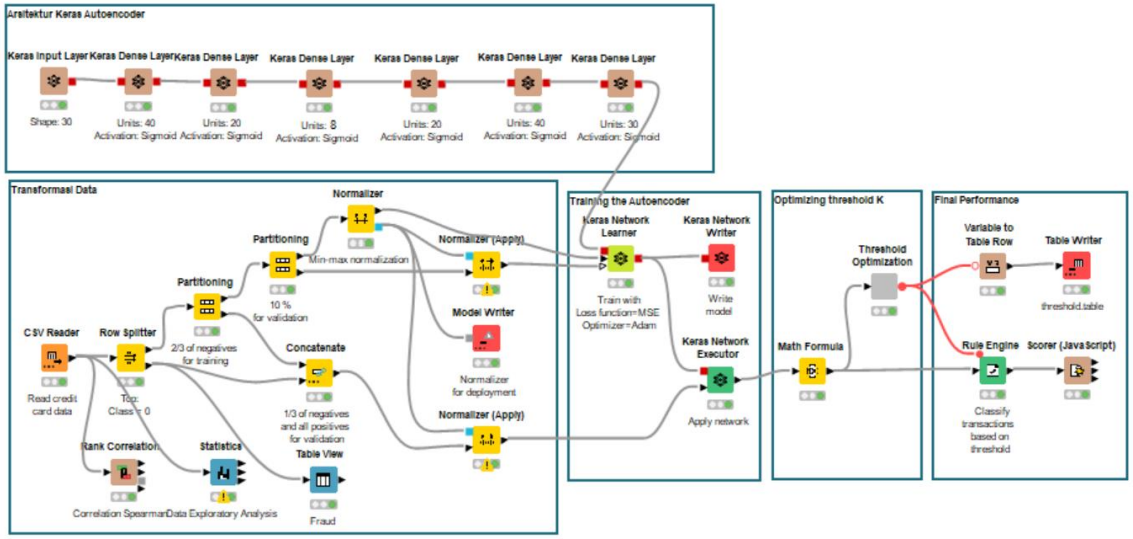
3.5 Neural Network Training

At the training stage, each weight and bias in each neuron will be updated continuously until the output is in line with expectations. Each iteration (epochs) will be evaluated to determine the stopping point of the machine learning process. The optimization algorithm in this study uses the Adaptive Moment Estimation (Adam) algorithm, which is part of gradient descent with stochasticity, which is widely used in computer vision and natural language processing (NLP).

Adam's algorithm updates the weighted network iteratively with the training data. The detail workflow can be found in Figure 8.

Figure 8. Augmentation of trained neural network using Adam Optimizer

Adam's algorithm configuration parameter uses optimizer settings on the KNIME platform with a learning rate or steps size for each update of 0.001. The smaller the value of the weight, the slower the machine to learn. Furthermore, Beta 1 represents the exponential decay rate for the



first-moment estimate, which is 0.9. While Beta 2 is used for the second exponential decay rate, which is 0.999. This value must be adjusted to close to 1.0 with a sparse gradient. Furthermore, epsilon is the smallest number used to prevent division by zero during implementation, for example, 10E-8.

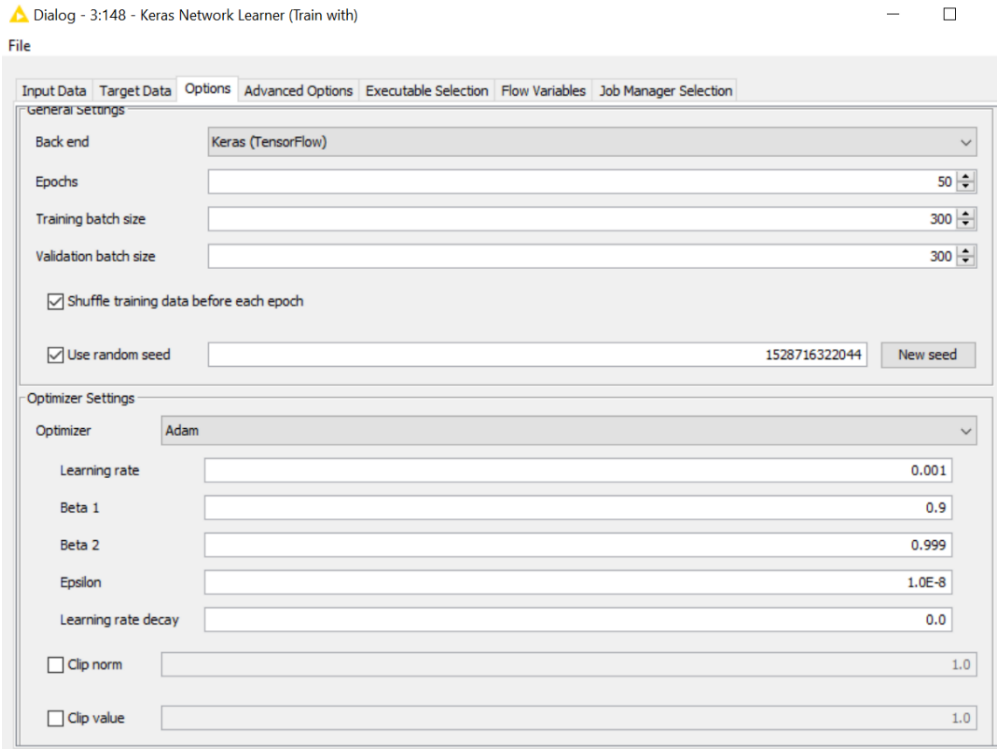
4 Results and Discussion

4.1 Model Testing and Results

The autoencoder was developed with 5 hidden layers, with 30-40-20-8-20-40-30 units with a sigmoid activation function. The KERAS input layer has nodes with a shape of 30. KERAS Dense

Layer requires a sigmoid to activate functions 40, 20, 8, 20, and 40 units. The Hard Dense Layer node will then define the output layer, with 30 units and sigmoid.

Figure 9. Overall model architecture and framework

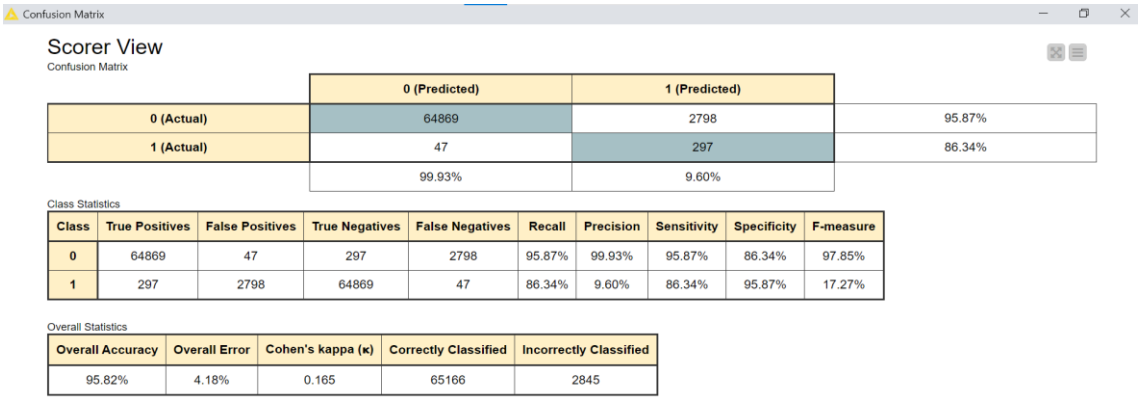


The model produces robust results showing model accuracy of 94.6% is given 50 epochs and 395 Batch. The iteration was conducted less than 6 minutes, showing the model required only less time-consuming. The detail can be found in Figure 9.

Figure 10. Model accuracy across epochs

Figure 11. Model loss across epochs

In addition to the accuracy, the model also demonstrates a loss of 0.0012 using typical epochs and batches. Subsequently, both parameters indicate that the model has better accuracy. The detailed result is depicted in Figure 10 and 11.



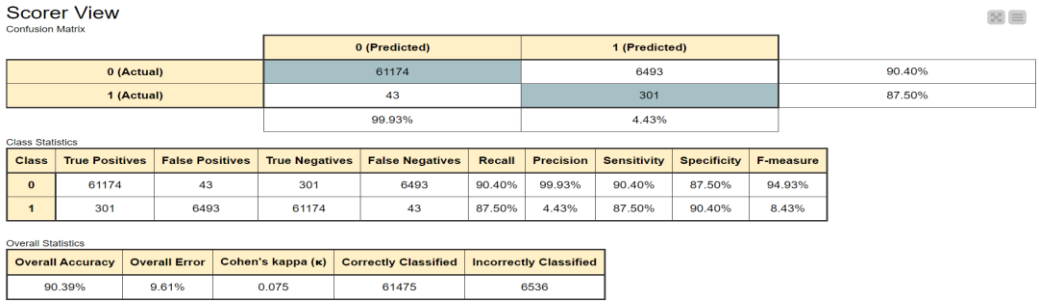
	0 (Predicted)	1 (Predicted)	
0 (Actual)	64869	2798	95.87%
1 (Actual)	47	297	86.34%
	99.93%	9.60%	

Class	True Positives	False Positives	True Negatives	False Negatives	Recall	Precision	Sensitivity	Specificity	F-measure
0	64869	47	297	2798	95.87%	99.93%	95.87%	86.34%	97.85%
1	297	2798	64869	47	86.34%	9.60%	86.34%	95.87%	17.27%

Overall Accuracy	Overall Error	Cohen's kappa (κ)	Correctly Classified	Incorrectly Classified
95.82%	4.18%	0.165	65166	2845

Figure 12. Overall model accuracy based on Adam Optimizer

The detailed result of an overall model can be depicted using a sort of model parameters found in Figure 12 (Adam optimizer) and Figure 13 (Adamax Optimizer). The measures include recall, precision, sensitivity, specificity, and F-measures. The model with Adam optimizer produces 95.82% overall accuracy and 0.165 of Cohen's kappa, which is relatively better than traditional machine learning found in the literature review discussed in the earlier section.



	0 (Predicted)	1 (Predicted)	
0 (Actual)	61174	6493	90.40%
1 (Actual)	43	301	87.50%
	99.93%	4.43%	

Class	True Positives	False Positives	True Negatives	False Negatives	Recall	Precision	Sensitivity	Specificity	F-measure
0	61174	43	301	6493	90.40%	99.93%	90.40%	87.50%	94.93%
1	301	6493	61174	43	87.50%	4.43%	87.50%	90.40%	8.43%

Overall Accuracy	Overall Error	Cohen's kappa (κ)	Correctly Classified	Incorrectly Classified
90.39%	9.61%	0.075	61475	6536

Figure 13. Overall model accuracy based on Adamax Optimizer

On the other hand, the model with Adamax optimizer produces lower accuracy of 90.38% and 0.075 of Cohen's kappa. Based on this comparative analysis, the Adam optimizer is then selected.

4.2 Model Selection

This section aims to deploy the selected model performed in the earlier section. The model and framework are used to implement the model using a new dataset. The results are shown in the Table view for the binary classes 0 and 1. The detailed workflow can be depicted in Figure 14.

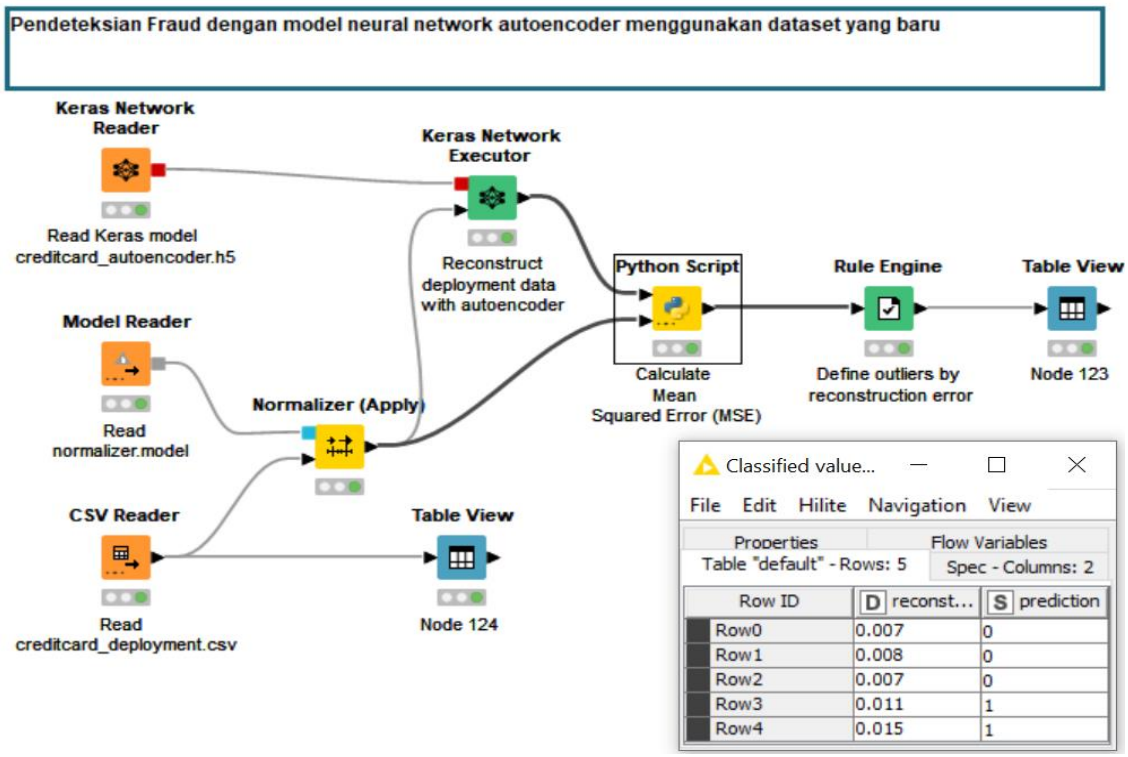


Figure 14. Deployment workflow for predicting the new testing set

According to the results found in Table 2, the algorithm enables us to detect 2 fraudulent transactions of 5 new datasets. The fraudulence transaction can be predicted (Pred. = 1) in a row number 4 and 5 of the new datasets while the first three rows are relatively legitimate transactions.

Table 2. Testing and result validation

ID	Amount	Var. 1	….	Var. 28	Error	Pred.	Res.
1	123.5	-0.966	….	0.061	0.007	0	L
2	69.99	-1.158	….	0.215	0.008	0	L
3	322.44	-0.854	….	0.003	0.007	0	L

4	937.69	-1.64	….	-0.41	0.011	1	F
5	179.66	-4.064	….	-0.631	0.015	1	F

F: Fraud; L: Legitimate

5. Conclusion

This study optimizes the machine mastering approach based totally on Neural Networks to improve model accuracy through the integration of KNIME and Python Programming with KERAS and TensorFlow models. The study additionally conducts a comparative analysis to scrutinize the overall performance of Adam and Adamax Optimizer. The model with Adam optimizer produces 95.82% overall accuracy and 0.165 of Cohen's kappa, which is relatively better than traditional machine learning found in the literature review discussed in the earlier section. The validation demonstrates that the algorithm enables us to detect fraudulent transactions.

REFERENCES

[1] Al-Hashedi, K.G., & Magalingam, P. (2021). Financial fraud detection applying data mining techniques: A comprehensive review from 2009 to 2019.

[2] Anggraeni, D., Sugiyanto, K., Zam Zam, M. I., & Patria, H. (2022). Stock Price Movement Prediction using Supervised Machine Learning Algorithm: KNIME. Junal Akun Nabelo: Jurnal Akuntansi, Netral, Akuntabel, Objektif. 4(2), 671–68.

[3] Asha, RB., & Kumar, S. Credit card fraud detection using artificial neural network. Global Transitions Proceedings. 2, 35-41 https://doi.org/10.1016/j.gltp.2021.01.006.

[4] Borgne, Y. L. & Bontempi, G. (2021). Machine Learning for Credit Card Fraud Detection – Practical Handbook.

[5] Johnson, J. M. & Khoshgoftaar, T. M. (2019). Survey on deep learning with class imbalance. Journal of Big Data, 6(27).

[6] Keras (2022). Adamax optimizer. https://keras.io/api/optimizers/adamax/.

[7] Mariana, CD., & Patria, H. (2021). Are Electronic Vehicle Stocks in ASEAN-5 Countries Investable during the Covid-19 Pandemic?Perceptions of Energy Resources Efficiency for Sustainable Development in the Developing Context of Nigeria: Implications for Enterprise Development in the Energy Sector. P. 184.

[8] Nilson report. Nilson report issue 1164. November 2019. [Online; Last consulted 09-October-2020].URL: https://nilsonreport.com/upload/content_promo/The_Nilson_Report_Issue_1164.pdf .

[9] Patria, H., & Adrison, V. (2015). Oil Exploration Economics: Empirical Evidence from Indonesian Geological Basins. Economics and Finance in Indonesia, 61(3), 196. https://doi.org/10.7454/efi.v61i3.514.

[10] Patria, H. (2021). The Role of Success Rate, Discovery, Appraisal Spending, and Transitioning Reion on Exploration Drilling of Oil and Gas in Indonesia in 2004–2015. Economics and Finance in Indonesia, 61(3), 196. https://dx.doi.org/10.47291/efi.v67i2.952.

[11] Patria, H. (2022). Predicting the Oil Investment Decision through Data Mining: Empirical Evidence in Indonesia Oil Exploration Sector. Data Science: Journal of Computing and Applied Informatics (JoCAI), 6(1), 1-11. https://doi.org/10.32734/jocai.v6.i1-7539.

[12] Zulfikri, F., Tryanda, D., Syarif, A., & Patria, H.(2021). Predicting Peer to Peer Lending Loan Risk Using Classification Approach. International Journal of Advanced Science Computing and Engineering, 3(2), 94–100. https://doi.org/10.30630/ijasce.3.2.57.