Global Academic and Scientific Journal of Multidisciplinary Studies (GASJMS)

Volume 4 | Issue 3, 2026 Homepage: https://gaspublishers.com/gasjms/ ISSN: 2583-8970

Effect of Automated Hyper-Parameter Tuning on Deep Belief Network for Credit Card Fraud Detection

Adeolu Johnson Olawale[1], Bosede Oyenike Oguntunde[2], Rafiu Adesina Ganiyu[3], Adeleye Samuel Falohun[3]

[1] *Department of Computer Engineering, Osun State University, Osogbo, Osun State, Nigeria*
[2] *Department of Computer Science, Redeemer's University, Ede, Osun State, Nigeria*
[3] *Department of Computer Engineering, Ladoke Akintola University of Technology, Ogbomoso, Oyo State, Nigeria*

Received: *01.03.2026* | ***Accepted:*** *23.03.2026* | ***Published:*** *25.03.2026*

***Corresponding Author:** Adeolu Johnson Olawale

DOI: *10.5281/zenodo.19222335*

Abstract **Original Research Article**

Credit card fraud detection (CCFD) is a critical facet in the financial industry as digital payments are becoming increasingly popular. While traditional rule based CCFD often fails due to concept drift and class imbalance challenges, Deep Belief Networks (DBN) can automatically learn complex hidden features but is computationally intensive and requires critical optimal hyperparameters tuning, which can be tuned with walrus optimization algorithm (WOA). This research therefore developed an automated hyperparameter tuning using walrus optimization algorithm to optimize deep belief network model (WT-DBN) for credit card fraud detection system. The research was carried out with dataset of a total of 10,000 real-world credit card transaction records. These data underwent pre-processing, after which a Walrus Optimization Algorithm was developed and applied to enhance the model detection capability by selecting the optimal hyperparameters. Finally, the WT-DBNs model was applied to credit card fraud detection, and implemented in MATLAB R2023a. At the highest training ratio of 80% for training and 20% for testing, Walrus tuned-DBN achieved a FPR, Sensitivity, Specificity, Precision, F1-Score, accuracy and detection time of 9.50, 97.5%, 90.5%, 97.62%, 96.86%, 96.1% and 25.91s, respectively as compared to 12.17 96.83%, 87.83%, 96.95%, 95.98%, 95.03% and 33.95s respectively for standard DBN. The best fitness value of Walrus tuned-DBN was 0.05883. The comparative results of the standard (DBN) and WT-DBN, across different data divisions shows that Walrus Optimized tuned-DBN(WT-DBN) enhances better convergence and improved classification performance compared to the conventional DBN.

Keywords: Algorithm, Optimization, Hyperparameter, Deep Belief Network, Credit card, Fraud detetction, Accuracy.

1. Introduction

Financial fraud in credit card denotes an unlawful or illicit usage of an account by individuals rather than the genuine owner, necessitating preventive strategies to curb such misuse. Analysing fraudulent behaviour aids in mitigating risks and safeguarding against future incidents. Specifically, financial fraud in credit card occurs after someone exploits someone else credit card aimed at individual gain without the owner's or issuing institution's knowledge (Kou et al., 2021).

Ongoing evolution of credit card fraud techniques necessitates continuous research and development of more advanced fraud detection models (Mienye & Jere, 2024). Financial institutions must invest in dynamic fake recognition scheme that leverage present analytics and artificial intelligence (Ryman-Tubb et al., 2018). Joint actions among banks, cybersecurity firms, as well as regulatory agencies is essential to wage war against deceitful transaction (Moyano & Ross, 2017). It is important to state that there is need for fraud recognition to be updated and adapted in order to meet up or mitigate financial losses, as cybercriminals comes up with several bad, non palatable tactics. The generation to come will detect fake transaction through integrating AI-driven solutions with regulatory frameworks to ensure secure and efficient financial transactions (Jurgovsky et al., 2018).

Detection of fake online transaction in credit card is now a significant issue which has gained weighty attention as a result of increasing sophistication of fraudulent activities (Narender & Anand, 2025). Traditional recognition methods, including rule-based systems as well as models in statistics, often struggle in coping with emerging fraud patterns (West & Bhattacharya, 2016). Deep learning approach have materialised as a significant alternative as a result of their capability in spontaneously extracting deep complex structures within huge financial datum (Liu et al., 2021). These scheme, which are Deep Belief Networks (DBNs), Convolutional Neural Networks (CNNs), and Recurring Neural Net (RNNs), really display effective methods to recognise deceit monetary communication (Goodfellow et al., 2016). The effectiveness of core computer model is in between its capability to work on serial data and recognise core deceitful scheme which previous model may not recognise.

Among deep learning approaches, DBNs have demonstrated promising results for recognising online deceit in credit card due to their hierarchical learning structure. Some layers of RBMs form a DBN that learn abstract representations of transaction data (Bengio, Y. 2009). This deep architecture enables DBNs to capture complex correlations between different transaction attributes, making them effective in fake recognition scheme. DBNs can learn feature depictions void of extensive physical feature application, as compared to previous scheme or models, (Zhai et al., 2018). So, with the aforementioned, they provide higher accuracy and generalization in detecting fraudulent transactions.

The efficacy of detecting financial deceits lies in the integration of advanced deep learning techniques, as time go by, such as DBNs, with real-time transaction monitoring systems (Mienye & Jere, 2024). Financial institutions must continuously update their deceit recognition scheme in order to meet up with dynamic deceits tactics (Ryman-Tubb et al., 2018). A combination of DBNs with reinforcement learning, transfer knowledge, with understandable AI could further improve deceits recognition accuracy and limpidity. Also, Collaboration between banks, cybersecurity firms, and regulatory agencies will be essential in developing robust fraud prevention strategies (Parra Moyano & Ross, 2017). As fraud techniques become more sophisticated, leveraging DBNs as well as remaining abysmal knowledge models will be important to minimize financial loss and ensuring secure transactions on financial data and customer money.

Conventional online deceit recognition models, including rule based systems as well as shallow system knowledge classifiers, could not properly address such inherent challenges of imbalanced datasets, concept Drift, real-time detection and Feature Engineering (Dal-Pozzolo et al., 2015), while Deep Belief Networks (DBNs) have demonstrated superior performance in detecting non-linear, low-frequency fraud patterns and possess the capability to autonomously derive complex structures within the premised of raw transactional dataset (Ghosh & Reilly, 2021). Their adoption is hindered by critical limitations which include high computational costs during training due to multi-layer architectures, requirement of manual meticulous hyperparameter tuning (such as, learning rates, layer sizes) which often lead to suboptimal convergence, overfitting on imbalanced datase, and instability in convergence when applied to highly imbalanced datasets (Goodfellow et al., 2016). Considering their weakness aforementioned, their sensitivity to weight initialization can degrade detection accuracy, predominantly within evolving places in which deceits methods used to come up frequently (Chen et al., 2022). Challenges aforementioned reduces DBN adaptability and application in real-world context requiring advanced optimization techniques to maximize system performance.

It is worth mentioning that when you improve these DBN parameters such as learning rates, number of layer neurons, regularization term, and layer configuration, WOA can reduce training time, minimize overfitting on imbalanced dataset, and improve detection accuracy. In this paper, we developed a walrus optimization algorithm to tune deep belief network hyper parameter for credit card fraud detection.

2. Methodology

The walrus tuned Deep Belief Network (WTBN) for credit card fraud detection was constructed through several phases as illustrated in Figure 1 below. First, a set of illegitimate and legitimate credit card transactions was collected from Sparkov Data Generation GitHub tool (SDGGT) at kaagle.com. The data was then normalized, empty spaces removed, and missing values filled to ensure data consistency. Next, a Walrus Optimization Algorithm (WOA) was created which was used to tune the optimal hyperparameters of Deep Belief Networks (WT-DBNs) during feature extraction. Finally, the walrus tuned-DBN was applied to detect credit card fraud detection, evaluating its sensitivity, specificity, precision, false positive rate, accuracy, F1-Score, and detection time for a robust and efficient fraud detection system.

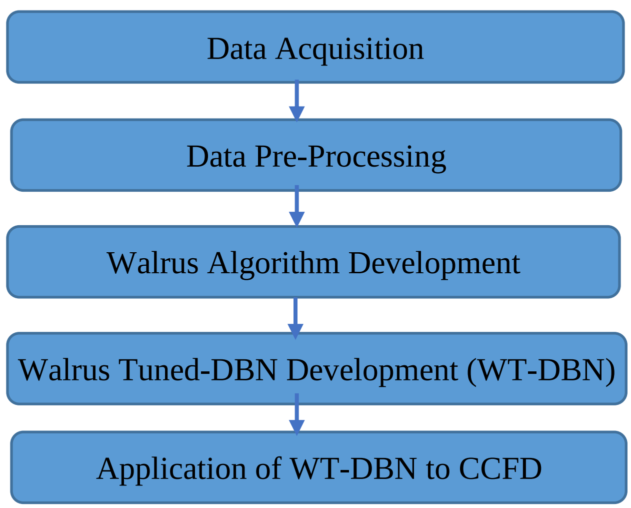

Figure 1: Schematic Diagram of the Developed WT-DBN

2.1 Data acquisition.

The dataset used for training the developed model were obtained through kaagle.com. the dataset consist of transaction records from a credit card transaction dataset which has both legitimate as well as fraudulent records starting from 1st of January, 2019, to 31st of December, 2020. It covers a pool of 800 merchants and 1000 customers' credit cards.

2.2 Data preprocessing.

Some inconsistencies such as missing values, duplicate entries, and un-normalized attributes, which may have a detrimental effect on model performance were treated. The preprocessing stage employed imputation techniques including mean, median, or mode for dealing with missing values. Min-Max Scaling was also applied to standardize values, one-hot encoding and label encoding, was employed to transform categorical variables (e.g transaction type and merchant location) into numerical formats while preserving their relationships. While Interquartile Range (IQR) was employed to identify and remove anomalous transactions that may not represent fraudulent activities.

2.3 Development of walrus optimization algorithm

The optimization algorithm begins with parameter initialization, where the optimization problem is defined as minF(X), X ∈ Ω, and the candidate solutions $X = [x_1, x_2, \ldots, x_n]$ are initialized. In the fitness evaluation phase, the WOA as shown in Algorithm 1 assess candidate solutions based on their fitness values. The strongest walrus, X*, is identified as the best solution, which serves as a reference for updating other walruses' positions. From algorithm 1, walrus movement follows fixed strategies of exploration (feeding strategy), migration (divergence maintenance strategy) and exploitation (survival from predator strategy). The update phase in WOA consists of three main movement strategies: feeding (exploration), migration (diversity maintenance), and escaping/fighting (exploitation) ensures better adaptability and allow the algorithm to explore new regions. Hence, the selection and stopping conditions ensure that the best candidate solution is retained.

Algorithm 1: Developed Walrus tuned-DBN

Step 1: Initialize Parameters

1. **Define the optimization problem:**

$$\min F(X), X \in \Omega$$

where:

$F(X)$ **is the objective function.**

$X = [x_1, x_2, \ldots, x_n]$ **represents candidate solutions.**

Ω **is the search space.**

2. **Set algorithm parameters:**

$N \rightarrow$ **Number of walruses.**

$T \rightarrow$ **Maximum number of iterations.**

$X_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,n}) \rightarrow$ **Initial walrus positions.**

Step 2: Fitness Evaluation

Compute fitness value for each walrus X_i **:**

$$F_i = F(X_i)$$

Identify strongest walrus X^* with the best fitness.

Step 3: Update Walrus Locations
Each walrus updates its position based on three main phases:
3.1 WOA: Original Algorithm

1. **Feeding Strategy (Exploration)**

Compute new position:

$$x_{i,j}^{y_n} = x_{i,j} + \text{rand}_{i,j} \cdot \left(X_j^* - I_{i,j} \cdot x_{i,j}\right)$$

Accept if fitness improves.

2. **Migration (Diversity Maintenance)**

- Choose new location:

$$x_{i,j}^{p_i} = \begin{cases} x_{i,j} + \text{rand}_{i,j} \cdot \left(x_{k,j} - I_{i,j} \cdot x_{i,j}\right), & F_k < F_i \\ x_{i,j} + \text{rand}_{i,j} \cdot \left(x_{i,j} - x_{k,j}\right), & \text{otherwise} \end{cases}$$

Migrate if the fitness improve

3. **Escaping and Fighting (Exploitation)**

Compute new position:

$$x_{i,j}^{x,} = x_{i,j} + \left(\text{lb}_{\text{iocel },j}^{t} + \left(\text{ub}_{\text{locel },j}^{t} - \text{ rand } \cdot \text{lb}_{\text{loceal },j}^{t}\right)\right)$$

Update if fitness improves.

Step 4: Selection of Best Solution

Evaluate new positions and update walrus locations

Store the best candidate solution

Step 5: Stopping Condition

If $t = T$, return the best solution: otherwise, repeat steps

2.4 Walrus Tuned-DBN Development (WT-DBN)

The turning process using the developed walrus optimization algorithm was applied to initialized and automate the set of DBN hyperparameters, which are the number of hidden layers, number of neurons per layer, learning rate, momentum, and weight decay. The WOA algorithm was initialized with a population of walruses, where each walrus represents a potential set of DBN hyperparameters. After initializing the DBN, the stacked Restricted Boltzmann Machines (RBMs) were pre-trained to extract hierarchical features from transaction data. The forward pass in RBM calculates hidden neuron activations using a sigmoid function as shown in equation (1), while the backward pass reconstructs the input data as shown in equation (2). This pre-training phase allows DBNs to capture meaningful representations from raw transaction data, making the subsequent classification more accurate, where the updates of weight in RBM were being achieved through Contrastive Divergence (CD) techniques in order to minimize reconstruction errors.

$$P\left(h_f = 1 \mid X\right) = \sigma\left(\sum_{i=1}^{n} w_{ij}x_i + b_j\right) \qquad (1)$$

$$P(x_i = 1 \mid H) = \sigma\left(\sum_{j=1}^{m} w_{ij}h_j + c_i\right) \qquad (2)$$

Where:

w_{lj} = weight between input i and hidden neuron j.

bj, c_i = bias for hidden and visible layer respectively.

2.5 Application of WT-DBN to CCFD

After the optimal DBN architecture is achieved through WOA, the model undergoes two training processes: unsupervised pre-training of stacked Restricted Boltzmann Machines followed by supervised fine-tuning via back-propagation. This training technique makes use of contrastive divergence to improve the weight update rate during the unsupervised phase and provides a robust feature representation that captures the subtle differences between genuine and fraudulent transactions. In the final classification step, we apply our model to new credit card transactions and give financial institutions an estimate of the likelihood of fraud, in order to achieves significantly higher detection rates while still offering lower false positive rates than traditional machine learning in detecting financial fraud.

2.6 Evaluation of the model Performance

In determining the performance of the WT-DBN model, some performance evaluation metrics were employed. These metrics measure or assess several facets of the model's capacity to correctly identify transactions as authentic or fraudulent. The metrics are False positive rate (FPR), sensitivity (recall), specificity, precision, accuracy, $\mathrm{F1-Score}$, and detection time as shown in equation (3) to (8) respective.

$$FPR = \frac{FP}{FP+TN} \times 100\% \qquad (3)$$

$$\text{Sensitivity} = \frac{TP}{TP+FN} \times 100\% \qquad (4)$$

$$\text{Specificity} = \frac{TN}{TN+FP} \times 100\% \qquad (5)$$

$$\text{Precision} = \frac{TP}{TP+FP} \times 100\% \qquad (6)$$

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \quad (7)$$

$$\mathrm{F1-Score} = \frac{2\times\mathrm{TP}}{2\times\mathrm{TP+FP+FN}} \times 100\% \quad (8)$$

$$\text{Detection Time} = \frac{\text{Total Processing Time}}{\text{Number of Transactions Processed}} \quad (9)$$

3. Result and Discussion

The results obtained in this study support the hypothesis that our algorithm to create automated tuned hyperparameter Deep Belief Network (WT-DBN) is effective at detecting credit card fraud. To test the effect of data split on detection performance, we considered four train-test splits of 60:40, 70:30, 75:25, and 80:20. From the result obtained, our TW-DBN model generated low False Positive Rate but high Sensitivity and Specificity across data divisions. The high Precision, F1-Score, and overall Accuracy show that our model can correctly identify fraudulent transactions as illegitimate and non-fraudulent transaction as legitimate ones even in lesser time.

In the Table 1 below, the iterative selection of tuned Deep Belief Network (DBN) hyperparameters with Walrus Tuned Algorithm were presented. The results show a progressive improvement in fitness values, indicating efficient convergence toward an optimal solution, where the fitness value reflects the classification error to be minimized. The minimum fitness value of 0.05883 was achieved at iteration 29, corresponding to a DBN configuration with 2 layers, 300 neurons, a learning rate of 0.02331, momentum of 0.87, and weight decay of 0.00111. This optimal

WT-DBN configuration was therefore selected for subsequent fraud detection experiments, as it provides the best trade-off between learning capability and error minimization.

Table 1: Selection of Optimal DBN hyperparameters with WOA

Iteration	Number of Layers	Number of Neurons	Learning Rate	Momentum	Weight Decay	Fitness Value
1	5	164	0.0217	0.602	0.00127	0.14612
2	6	80	0.01425	0.624	0.00646	0.16074
3	4	118	0.04501	0.601	0.00482	0.15057
4	4	203	0.03947	0.858	0.00667	0.27222
5	4	182	0.01955	0.675	0.00831	0.09282
6	5	159	0.02906	0.889	0.00941	0.11768
7	3	108	0.04418	0.688	0.00981	0.27306
8	4	272	0.01668	0.783	0.00425	0.14241
9	6	130	0.03278	0.765	0.00777	0.06066
10	6	52	0.00576	0.602	0.00283	0.20868
11	5	209	0.02471	0.691	0.00059	0.11211
12	4	175	0.00704	0.762	0.00447	0.17715
13	2	170	0.04741	0.701	0.00574	0.28524
14	3	101	0.00103	0.732	0.00441	0.09385
15	6	222	0.01266	0.697	0.00032	0.2115
16	6	146	0.01254	0.934	0.00358	0.13321
17	4	114	0.02081	0.763	0.0082	0.10884
18	5	139	0.04485	0.819	0.00568	0.27983
19	3	109	0.01069	0.614	0.00351	0.26897
20	4	165	0.04761	0.855	0.00575	0.18965
21	4	156	0.00074	0.887	0.00823	0.15501
22	3	211	0.02426	0.89	0.0083	0.10911
23	2	165	0.01021	0.816	0.00581	0.28531
24	5	244	0.04484	0.729	0.00946	0.14856
25	6	53	0.03708	0.714	0.00608	0.15054
26	6	202	0.02743	0.8	0.00325	0.25483
27	6	101	0.04636	0.814	0.00256	0.14129
28	4	79	0.01082	0.623	0.00206	0.12566
29	**2**	**300**	**0.02331**	**0.87**	**0.00111**	**0.05883**
30	4	141	0.04932	0.77	0.00281	0.26283

3.1 Comparison Result of DBN and WT-DBN Model

Table 2 presents the comparison results of the standard DBN and Walrus Tuned-DBN (WT-DBN) for different divisions of data. The standard DBN employed manually chosen hyperparameters, which are less adaptable to complex frauds. But, the WT-DBN exploits the walrus optimization algorithm to automatically select the appropriate hyperparameters of the regular DBN. The hyperparameter selected or tuned with walrus optimization are number of layers, number of neurons, learning rate, momentum, and weight decay, which will refine DBN to achieve better convergence and classification performance than the conventional DBN.

Considering 60×40 data division, the false positive rate (FPR) got declined to 5.42% for WT-DBN whereas it remains 6.67% for DBN, demonstrating superior discrimination of legitimate transactions. Sensitivity and specificity values of 96.83% and 94.58% for WT-DBN demonstrate more accurate fraud detection and reduced misclassification. Furthermore, WT-DBN realized the greater accuracy of 95.93% with the shorter detection time of 25.02 seconds. These results clearly show the benefit of applying walrus optimization algorithm for hyperparameter tuning.

Similar performance trends were observed for the 70×30 data split. The sensitivity of WT-DBN reached 97.24% while that of DBN is at 96.48%, confirming improved hyperparameter tuning. A lower false positive rate of 6.89% and higher specificity of 93.11% further highlight its robustness. The accuracy of WT-DBN improved to 96%, while detection time dropped to 25.91 seconds, demonstrating computational efficiency.

For the 75×25 data division, WT-DBN maintained better performance with an accuracy of 96.07% and an F1-score of 96.72%, as against DBN which recorded 94.97% of accuracy and 95.79 of F1-Score, indicating the advantage of optimization-based hyperparameter tuning. The reduced false positive rate of 7.87% for WT-DBN confirms improved generalization ability. This highlights the effectiveness of enhancing the deep belief network by automating its hyperparameter selection or tuning.

At the highest training ratio of 80×20, WT-DBN demonstrated notable improvements, with an accuracy of 96.1% and the highest sensitivity of 97.5% as compared to accuracy of 95.03% and sensitively of 96.83% of ordinary DBN demonstrating higher fraud detection capability of WT-DBN model. The false positive rate of 9.50% remained significantly low as compared to that of DBN, emphasizing reliable fraud detection under larger training sets. Overall, the comparative results confirm that DBN hyperparameters tuning using walrus optimization algorithm yields superior accuracy, FPR, Sensitivity, Specificity, Precision, F1-Score, Robustness, and detection speed compared to the conventional DBN model.

Table 2: Combined Evaluation Result based on DBN and WT-DBN

Techniques	Data Division	FPR (%)	SEN (%)	SPEC (%)	PREC (%)	F1-SCORE (%)	ACC (%)	Time (sec)
WT-DBN	60x40	5.42	96.83	94.58	96.4	96.17	95.93	26.02
DBN		6.67	95.94	93.33	95.57	95.24	94.90	34.12
WT-DBN	70x30	6.89	97.24	93.11	97.05	96.52	96	25.91
DBN		8.67	96.48	91.33	96.29	95.61	94.93	33.95

WT-DBN	75x25	7.87	97.38	92.13	97.38	96.72	96.07	26.05
DBN		10.13	96.67	89.87	96.62	95.79	94.97	33.95
WT-DBN	80x20	9.50	97.5	90.5	97.62	96.86	96.1	26.01
DBN		12.17	96.83	87.83	96.95	95.98	95.03	33.97

Figure 2 below shows a bar charts displaying the metrics performance of the two models discussed previously while Figure3 shows how effective tuning of hyperparameter for deep belief network can reduce the detection time across all data divisions of training to testing ratios, demonstrating the ability of well-tuned DBN to address the challenges of real time applications like that of credit card fraud detection applications.

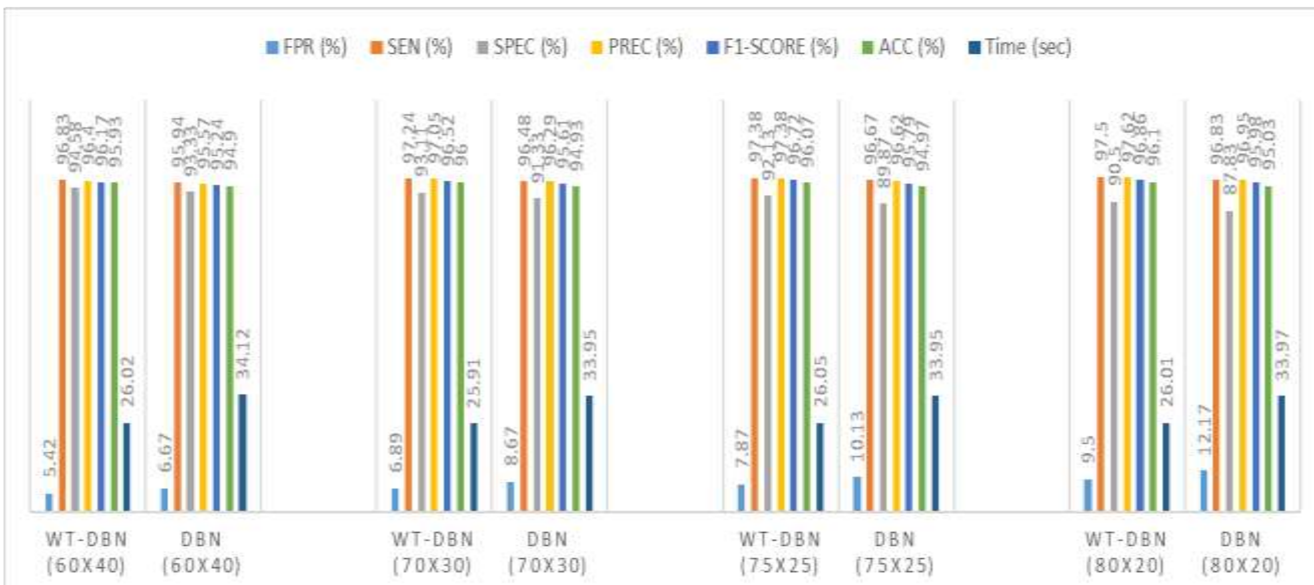

Figure 2: Performance metrics Display

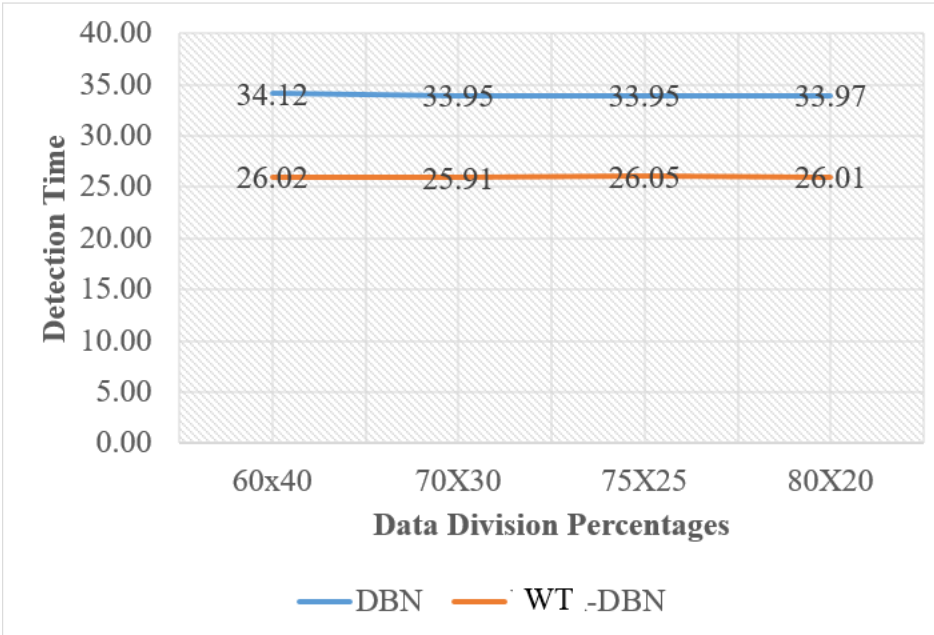

Figure 3: Detection time Comparison

4. Conclusion

The study used the developed walrus optimization algorithm to improve the DBN hyperparameter for effective deep belief network model to detect fraud in credit card. The WT-DBN is more robust and convergence faster across different search landscapes, which provided a reliable and efficient framework for optimizing complex, high-dimensional models such as the Deep Belief Network (DBN) in the context of fraud detection. The WOA formulation was used to make better the hyperparameters and network architecture of the DBN into the WT-DBN model. Simulation of the developed WT-DBN model was conducted in the Matlab R2023a software. We compared the performance of the WT-DBN model with standard DBN models by using a comprehensive set of metrics such as False Positive Rate (FPR), Sensitivity, Specificity, Accuracy, Precision, F1-Score, Average Detection Time and the result shows that WT-DBN perform far better in terms of realizing lower FPR, exhibiting greater fraud detection capability and achieving reduction in fraud detection time.

Acknowledgement

We are grateful to the Lord almighty and the author of all knowledge. Also, we are grateful to the management of Ladoke Akintola university of Technology, Ogbomoso and the management of Osun State University Osogbo, for creating enabling academic environment to carry out this research.

References

Kou, G., Xu, Y., Peng, Y., Shen, F., Chen, Y., Chang, K., & Kou, S. (2021). Bankruptcy prediction for SMEs using transactional data and two-stage multiobjective feature selection. *Decision Support Systems*, *140*, 113429. https://doi.org/10.1016/j.dss.2020.113429

Mienye, I. D., & Jere, N. (2024). Deep learning for credit card fraud detection: A review of algorithms, challenges, and solutions. *IEEE access*, *12*, 96893-96910.

Ryman-Tubb, N. F., Krause, P., & Garn, W. (2018). How Artificial Intelligence and machine learning research impacts payment card fraud detection: A survey and industry benchmark. *Engineering Applications of Artificial Intelligence*, *76*, 130-157. https://doi.org/10.1016/j.engappai.2018.07.008

Parra Moyano, J., & Ross, O. (2017). KYC optimization using distributed ledger technology. *Business & Information Systems Engineering*, *59*(6), 411-423.

Jurgovsky, J., Granitzer, M., Ziegler, K., Calabretto, S., Portier, P. E., He-Guelton, L., & Caelen, O. (2018). Sequence classification for credit-card fraud detection. *Expert systems with applications*, *100*, 234-245.

Narender, M., & Anand, A. J. (2025). Artificial Intelligence in Financial Fraud Detection. In *Handbook of AI-Driven Threat Detection and Prevention* (pp. 193-207). CRC Press.

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

Liu, S., Liu, L., Chawla, S. & Sahay, S. (2021). Fraud detection using machine learning and statistical models. *Computers & Security*, *105*, 102-256.

Goodfellow, I., Bengio, Y. & Courville, A. (2016). Deep feedforward networks. *Deep learning*, *1*, 161-217.

Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, *2*(1), 1-127.

Zhai, S., Cheng, Y., Lu, W. & Zhang, Z. (2018). Deep belief networks for feature extraction from credit card transaction data. *Neural Networks*, *98*, 110.

Dal-Pozzolo, A., Caelen, O. & Bontempi, G. (2015). When is undersampling effective in unbalanced classification tasks? *European Conference on Machine Learning*, 200–215.

Ghosh, S. & Reilly, D. L. (2021). Deep learning approaches for anomaly detection in transactional data. IEEE Transactions on Neural Networks, *32*(7), 3120–3131.

Chen, G., & Zhang, J. (2022). Applying Artificial Intelligence and Deep Belief Network to predict traffic congestion evacuation performance in smart cities. *Applied Soft Computing*, *121*, 108692.