Yanbu Journal of Engineering and Science

Volume 20 | Issue 2 Article 7

2023

Scalable Network Intrusion Detection in Cloud Environments through Parallelized Swarm-Optimized Neural Networks

Ramakrishnan Ramamoorthy
Yanbu Industrial college, Saudi Arabia

Ramesh Kumar Ranganathan
Yanbu Industrial college, Saudi Arabia

Sivakumar Ramu
Yanbu Industrial college, Saudi Arabia

Follow this and additional works at: https://yjes.researchcommons.org/yjes

The journal in which this article appears is hosted on Digital Commons, an Elsevier platform.

Recommended Citation

Ramamoorthy, Ramakrishnan; Ranganathan, Ramesh Kumar; and Ramu, Sivakumar (2023), Scalable Network Intrusion Detection in Cloud Environments through Parallelized Swarm-Optimized Neural Networks, *Yanbu Journal of Engineering and Science*: Vol. 20: Iss. 2, 62-70.
DOI: https://doi.org/10.53370/001c.92140
Available at: https://yjes.researchcommons.org/yjes/vol20/iss2/7

This Research Article is brought to you for free and open access by Yanbu Journal of Engineering and Science. It has been accepted for inclusion in Yanbu Journal of Engineering and Science by an authorized editor of Yanbu Journal of Engineering and Science.

Ramamoorthy R, Ranganathan RK, Ramu S. Scalable Network Intrusion Detection in Cloud Environments through Parallelized Swarm-Optimized Neural Networks. *YJES*. 2024;20(2):62-70. doi:10.53370/001c.92140

Research article

Scalable Network Intrusion Detection in Cloud Environments Through Parallelized Swarm-Optimized Neural Networks

Ramakrishnan Ramamoorthy[1], Ramesh Kumar Ranganathan[2], Sivakumar Ramu[2]

[1] Computer Science and Engineering , Yanbu Industrial College, [2] Computer Science and Engineering, Yanbu Industrial College

Keywords: Cloud computing, Network traffic, Network intrusion detection, Cloud environment

https://doi.org/10.53370/001c.92140

Yanbu Journal of Engineering and Science

Vol. 20, Issue 2, 2023 Submitted: 11 OCT 2023 Revised: 03 NOV 2023 Accepted: 26 NOV 2023

Cloud computing (CC) offers on-demand, flexible resources and services over the internet, to secure cloud assets and resources, privacy and security remain a difficult challenge. To overcome this problem, we proposed a Modified Dove Swarm Optimization Based Enhanced Feed Forward Neural Network (MDSO-EFNN) to examine the network traffic flow that targets a cloud environment. Network Intrusion detection systems (NIDSs) are crucial in identifying assaults in the cloud environment, which helps to reduce the problem. In this study, we gather an NSL-KDD network traffic dataset.

Secondly, collected data is preprocessed using Z-Score normalization to clean the data. Thirdly, Continuous wavelet transform (CWT) is employed to extract the unwanted data. Ant colony optimization (ACO) is used to choose the appropriate data. The selected appropriate data is used to test the process using MDSO-EFNN. The simulation findings of the result use a Python tool. As a result, our proposed method achieves significant outcomes with classification of accuracy (95%), precision rate (97%), sensitivity (98%), and specificity (96%).

1. INTRODUTION

Cloud computing (CC) paradigm is a cutting edge approach to computing that offers the consumers access to resources and applications as a service over the Internet, meeting their computing needs.[1] It offers Infrastructure as a Service (IaaS) and Software as a Service (SaaS), Platform as a Service (PaaS) influences the availability, confidentiality and integrity of cloud resources, major cloud providers such as Windows Azure, Rack Space, Eucalyptus, Open Nebula, and Amazonetc.Implements a firewall to guard against the incursions on cloud services.[2] Comparing network security in cloud computing and self-defense depends on the intrusion detection systems. Network Intrusion Detection Systems (NIDS) which is based on data collected from the network. The system analyzing traffic captures and general data acquired to identify malicious activity and intrusion attempts inside network.[3] These are examined and contrasted with a set of guidelines for recognizing attack signatures. Recorded traffic or security logs may contain patterns of traffic that indicates certain types of assaults. An intrusion detection system analyzes this data to detect assaults and harmful activity. Enormous amounts of data analyzed to identify patterns that match the signature stored in database.[4,5] The process requires rapid access to data storage and significant computational power in effective execution throughout extensive networks. The intrusion detection system (IDS) is spread over a vast area which includes many IDSs such as HIDS, NIDS or a combination of network and collaborating to observe and detect its occurrences.[6]According to the method used for intrusion detection occurs, three detection techniques exist, hybrid, and signature and anomaly-based, IDS detects intrusions using an anomaly approach by examining the system's aberrant behavior patterns events on a network or activity.[7]In the case of signature based ID, an IDS searches for known attack patterns by matching signatures. NIDS is configured at the entry point of a cloud network and connected to the switches that link the networks of the cloud servers or virtual machines.[8] NIDS audits updated the flow of traffic that enters into the cloud networks. Figure 1 shows a cloud environment.

Finding anomalies: The identification of deviations in traffic is the foundation of approach to optimizing transportation systems. This approach measures the deviation of observed traffic from its typical profile and there have been many implementations of this method using various parameters to quantify the divergence in the traffic profile.

A paradigm of shared responsibility is used for cloud computing security.[9] The security of the cloud's underlying infrastructure, which includes physical data centers, network infrastructure and hypervisors is the responsibility of cloud service providers (CSPs) also known as cloud service providers.[10] Concurrently, consumers of cloud services are responsible for ensuring the safety of their data, app and configurations to store in the cloud.

The main objective is to detect possible security risks inside cloud-based infrastructures, such as cyber-attacks and illegal access and to reduce the impact of threats. To improve the overall security posture of companies that depend

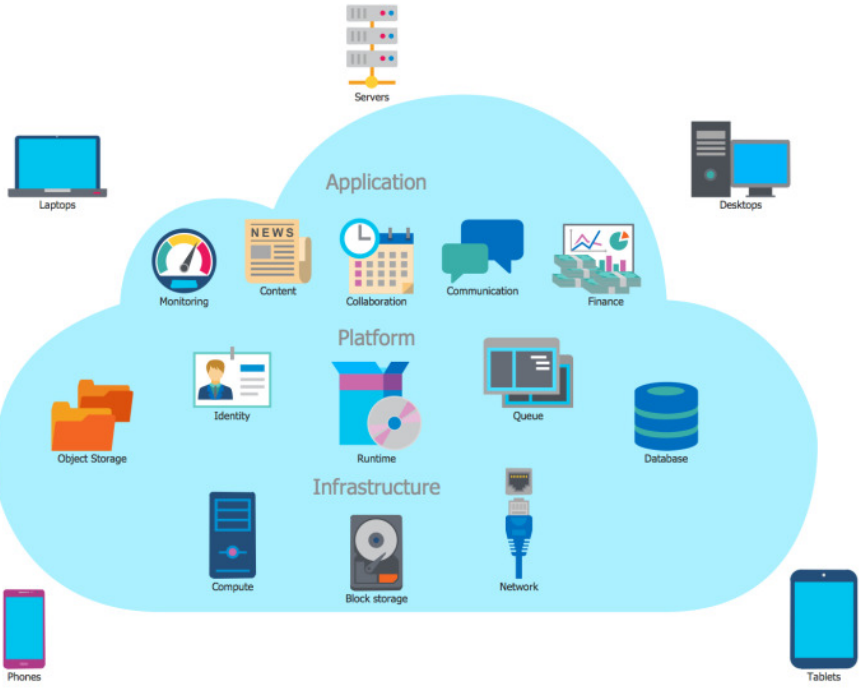

Fig. 1. Structure of Cloud Environment

on cloud services and resources, it is necessary to address the specific problems that are provided by cloud computing.

2. RELATED WORK

The research[11] described GL based cloud intrusion detection solution to solve neural network challenges. Through training efficiency, the proposed intrusion detection model enhances detection accuracy. The suggested model outperformed standard methods with 97% detection rate, 92% accuracy, and 91% precision. The study[12] explained intrusion detection model was built using the random forest technique. Real time network traffic data of each cloud server's various network levels has gathered using the tcp dump tool and data mining technologies. The experimental findings, device was capable of reducing the training and test times for intrusion detection and its detection accuracy can reach up to 93%. The article[13] described to development of the DNN architecture, which aims to develop the protection of cloud IDS systems. The effectiveness of the proposed DNN has assessed using the present approach. The results of the simulation test shows that the recommended terms of F-Score, AUC, FPR, accuracy, detection rate, and precision. The article[14] identified an intruders and neutralized assaults in the cloud paradigm proposed the improved invasive weed optimization .It was more practical to detect with a maximum accuracy 97% respectively. The study[15] described SVM classifier has applied and its approach for the purpose of classifying network data into normal and pathological patterns of activity determining the proposed system. The findings of the experiments demonstrated that the proposed system has capable of identifying aberrant behaviors with high degree of accuracy.

The research[16] proposed a lambda based edge cloud deep IDS approach for IoT security. Solution decreases training phase time and increases true positive assault detection accuracy compared to standard ML System detects real-time suspicious behavior and classifies it using batch analysis of historical data. The research[17] proposed a unique hybrid intrusion recognition structure based on ML and genetic algorithm approaches were integrated with a novel suitability created to assess system accuracy. The results indicated as 87% ofthe recommended model performed in comparison to standards. An anticipated is a higher level degree of symmetry in relation to identification of invasions and information securityof harmful activity, which makes this system successful in cloud computing. The article[18] explained to convolutional deep learning approach in addition to supervised the methods such as SVM and KNN models, the development of class methods is known as intrusion detection systems (IDSs) which aims to monitor and safeguard a network's resources.

The study[19] presented effective cluster-based IDS to resist online and cloud computing attacks setting accuracy, recall and F1-score was evaluated between suggested and current systems. The study[20] proposed approach detects cloud invasions using a sparse auto encoder, Bi-directional LSTM, dropout layer, and attention mechanism. Based on simulations, the proposed model has 98% precision, 99% recall, and above 98% accuracy. The article[21] described a technique for cloud computing DDoS attack detection. Reducing misclassification errors in DDoS detection is the main goal of this paper. The mutual information of RF importance approaches are two feature selection strategies that use in the proposed study to choose the most relevant features. The experimental findings demonstrated that the random forest with 19 characteristics have an accuracy

(97%). The paper[22] suggested a novel IDS used k means clustering and optimum fuzzy logic. The input dataset is grouped into clusters; recall and F-measure are evaluated. The proposed strategy outperformed others based on the findings. The article[23] explained strategy for network intrusion detection that is based on DL proposed method makes use of many different classification algorithms, such as the Auto Encoder, Long-short-term-memory Multi-Layer Perceptron, finding results are proposed approach detects abnormalities with great precision and minimal incidence of false alarms. The study[24] explained the design of Software networking defensive organization for online attack recognition and improvement. Anomaly detection and mitigation comprise the defense system. The detection model detects traffic abnormalities using CNN. By sending flow rule directives from the controller, the mitigation model filters abnormal traffic and traces the attacker through IP. Experimental results prove the SDN defense system's real-time DDoS attack flow detection and mitigation accuracy.

3. METHODOLOGY

The suggested techniques are used to fulfill the list of activities that follows: First we collected the dataset and preprocessed the data using Z-score normalization. Then we utilize continuous wavelet transform to extract features from the preprocessed data, and finally, we perform feature selection using Ant colony optimization (ACO).To detect the network traffic by using Modified Dove Swarm Optimization based Enhanced Feed Forward Neural Network (MDSO-EFNN). Figure 2 demonstrates the flow of the suggested approach.

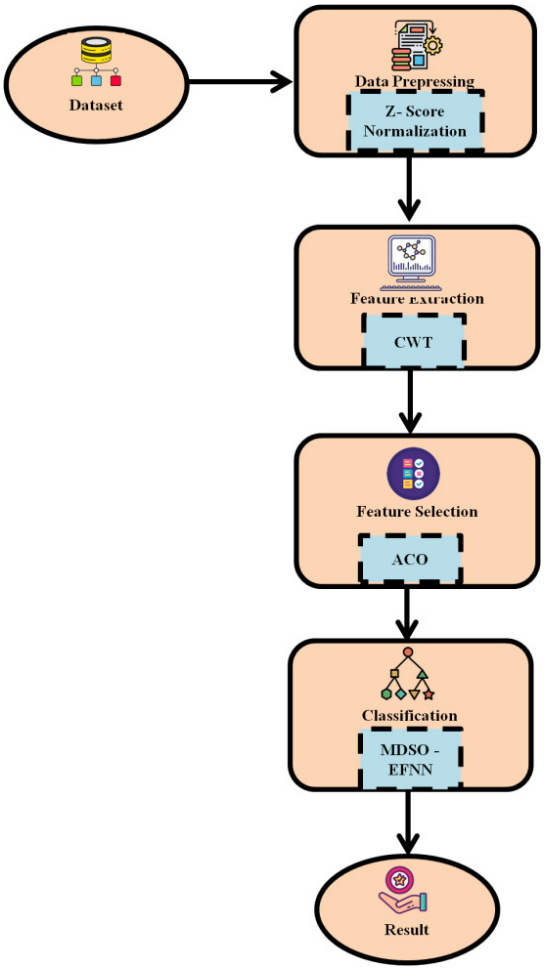

Fig. 2. Flow Diagram of Methodological design

3.1. DATASET

The data instances' observed labels were tested using the NSL-KDD test dataset. Performance is evaluated by comparing observed labels to the anticipated labels after classification.[25]Attacks against networks are very frequent, since the users has weak usernames and combinations of passwords to use nine algorithms, for the purpose of categorization. To NSL-KDD recommended as a way to address the issues in the original dataset KDD99. The high number of duplicate records in the KDD data collection is one of the biggest faults, since it favors frequent records, it influences the learning algorithms.

3.2. DATA PREPROCESSING

Z-score describes the procedures carried out to prepare a dataset for utilization the statistical process of standardization. This process also known as Z-score normalization itused to fix a dataset mean and standard deviation to 0 and 1 respectively. There is a bias toward the certain network properties over others, since network connections have numerous attributes with a wide range of values. Addressing the bias issue in Network Intrusion Detection System (NIDS) can be achieved by normalizing the characteristics using a specific normalization technique, which scales the feature values of connection instances to a predetermined range. This process enhances the NIDS's by transforming the data into a standard form. Raw data gathered from various input sources can be challenging. As a result, specific techniques are utilized to turn data into a dataset. This technique is known as data pre-processing when normalized data is used without performing a value analysis to improve the results.

In the pre-processing of a normalized database, missing values can be substituted using Z-scores. Normalization with the same standard deviation is employed to reduce rating variances with databases. This provides the amount of standard deviations given a data point deviates from the mean. The patient data is organized into a vector matrix with'm' columns and 'n' rows, where'm' is the number of features and 'n' is the number of patients. Commonly, it is used to standardizethe variables with various scales or measurement units. The Z-score normalizing entails deducting the dataset's normal from each data point and dividing the outcome by the normative variation.

$$(Z-score)_V = \frac{(V_{th}-\mu)}{\sigma} \quad (1)$$

According to the aforementioned equation the maximum allowable Z-score is calculated as the difference between the mean of the features and their standard deviation.

3.3. FEATURE EXTRACTION

In feature extraction using CWT, the signal is first converted into a time-scale representation and then the relevant information for the transformation is extracted from this representation.

3.3.1. CONTINUOUS WAVELET TRANSFORM

A CWT is referred as CWT;itis an effective method for analyzing temporal and frequency information. CWT is capable of localized decomposition in the use of techniques, such as the short-time Fourier transform. In addition, the mother wavelet function has a scale parameter that can be modified and it has a translation parameter, which enables it to divide a signal into multiple resolutions. The time-scale waveform family is made up of two parameters that have been adjusted, and it is represented by the following:

$$\varphi_{b,a}(s) = \frac{1}{\sqrt{b}}\varphi\left(\frac{s-a}{b}\right) \tag{2}$$

$$d(b,a) = \frac{1}{\sqrt{b}}\int_{-\infty}^{+\infty} w(s)\varphi^*\left(\frac{s-a}{b}\right)cs \tag{3}$$

$$d(b,a) = \frac{1}{\sqrt{b}}\int_{-\infty}^{+\infty} G(s)\varphi^*\left(\frac{s-a}{b}\right)cs \tag{4}$$

Translation parameter and c for the analytic function known as the mother wavelet function, provided signal G(s) wavelet transform. Thewavelet coefficients at scale and translation b are represented by w (s) and ψ* indicates the complex conjunction of ψ.

3.4. FEATURE SELECTION

An intelligent technique called ACO is used in NIDS to enhance the detection accuracy and minimize while boosting security.

3.4.1. ANT COLONY OPTIMIZATION (ACO)

To identify intrusions using clustering-based ACO (Ant Colony Optimization), network data records, which consist of various properties, must be treated as objects. Based on ACO these items are categorized into distinct types by intrusion detection. They vary in kind and typical intrusions. In the process training stage data is clustered and predetermined, number of clusters according to the similarity index. Every cluster's center is determined by averaging the items, that is a part of the group and hub of the courses inside data that embody the model is determined as the cluster centers average for the class. Considering a single class may include several clusters. These types of model are used as intrusion detection classifiers.

3.5. MODIFIED DOVE SWARM OPTIMIZATION (MDSO)

The cooperative behavior of doves in nature serves the novel method, which is modified to optimize the design and parameters of EFNN, during the training phase, MDSO-EFNN uses a population-based method inspired on thedove swarms, to adjust neural network architectures and improve the biased and weighted characteristics of its neurons

The doves graze in public areas when crumbs are scattered about and how hunts for crumbs dove. Not all doves are content. Unsatisfied doves may fly ahead in search of more crumbs. It notices that feed doves may inhabit crumb rich places. Proposed a whole different ideal algorithm as a result of the dove eating patterns, the optimization goal function in this approach is $l(U)$. The number of bread crumbs at given location, denoted by the symbol U, in a data collection is indicated by the symbol$l(U)$. The MDSO algorithm's flowchart is shown in Figure 3. Assume that N is the predetermined number of doves; it is recommended to arrange the doves in a regular square or rectangular design.

THE MDSO ALGORITHM

Step 1: Select the quantity of doves to release onto the solution area. Let assume that N is the prescribed amount of doves.While arranged in any random pattern around the area, to advise placing the doves consistently around a rectangle area.

Step 2: The position vector of dove d to be initialized using one of two techniques, Initializing Lattice Initialized is an alternative method. To streamline the process of creating a topologically ordered feature map, two effective influence initializations and suggested to set the weight vectors. Assume that the parameter space's smallest hyper rectangle, which includes all of the parameters' valid values.

Step 2.1 Initializing of the cells on the four vertices: each corner of the network's weight vectors are initialized as (1)

$$\begin{aligned}
&\underline{x}_{1,1} = (k_1, k_{2,\ldots,}, k_N)^S \\
&\underline{x}_{B,A} = (V_1, V_{2,\ldots,}, V_N)^S \\
&\underline{x}_{1,A} = \left(k_1, k_{2,\ldots,}k_{\lfloor N/2\rfloor}), {}^r v_{\lfloor N/2\rfloor}\right) + 1, \ldots, v_N\Big)^S \\
&\underline{x}_{B,1} = \left(v_1, v_{2,\ldots,}v_{\lfloor N/2\rfloor}\right), {}^n k_{\lfloor N/2\rfloor}\Big) + 1, \ldots, k_N\Big)^S
\end{aligned} \tag{5}$$

Step 2.2 .To initialize the cell values along the four edges as follows: (2)

$$\begin{aligned}
\underline{x}_{B,1} &= \frac{x_{B,A} - \underline{x}_{B,1}}{A-1}(I-1) + \underline{x}_{B,1} \\
&= \frac{i-1}{A-1}\underline{x}_{B,A} + \frac{A-i}{A-1}\underline{x}_{B,1} \text{ for } i = 2, \ldots, B-1
\end{aligned} \tag{6}$$

It collects the difference value and highest number for each neuron in order to assess the training performance. The following represents the learning rate's decreasing rate from its original value of 0.1.

Step 3: Determine each dove's fitness function, total number of crumbs at the dove's position.

$$\begin{aligned}
\underline{x}_{j,A} &= \frac{\underline{x}_{B,A} - \underline{x}_{1,A}}{B-1}(j-1) + \underline{x}_{1,A} \\
&= \frac{j-1}{B-1}\underline{x}_{B,A} + \frac{A-i}{A-1}\underline{x}_{1,A} \text{ for } i = 2, \ldots, B-1
\end{aligned} \tag{7}$$

Step 4:Using the greatest criteria at epoch, find the dove closest to the most crumbs.

$$\eta(m) = \eta 0 \times \left(1 - \frac{s}{S}\right) = 0.1\left(1 - \frac{s}{100}\right) \tag{8}$$

Step 5:Utilizing the following formula, update the degree of satiety for each dove.

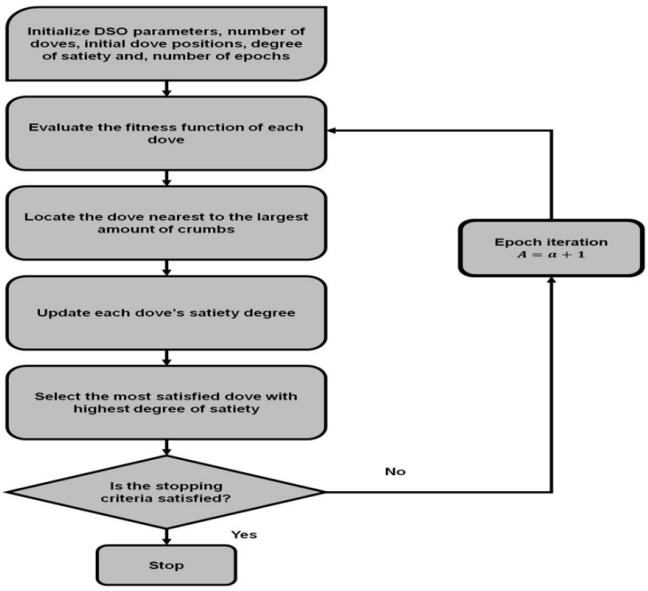

Fig. 3. Flow of Modified Dove Swarm Optimization

$$c_i^f = argmax\ \left\{e\ \left(x_i^f\right)\right\},\ \ for\ i = 1, \ldots, M \tag{9}$$

Step 6:Using the following maximum criteria, choose the dove that is the most pleased and has reached the greatest level of satiety.

$$T_i^f = \lambda T_i^{f-1} + f^{(e(x_i) - e\,(x_{ce})),\quad for\ i=1,\ldots,M} \tag{10}$$

Step 7:Utilizing the following maximum criteria, update the position vector of each dove.

$$\underline{x}_i^{f+1} = \underline{x}_i^f + \eta\beta_i^f(\underline{x}_{c_t}^f - \underline{x}_i^f) \tag{11}$$

Step 8:Proceed to step 3 and keep increasing the number of epochs with one ($a = \alpha$+1) until the termination clause is satisfied. The following is the terminate clause.

$$\beta_i^f = \left(\frac{T_{a_s}^f - T_i^f}{T_{a_s}^f}\right)\left(1 - \frac{\left\|\underline{x}_i^f - \underline{x}_{c_t}^f\right\|}{maxDistance}\right) \tag{12}$$

3.6. FEED FORWARD NEURAL NETWORK

The term feed forward neural network refers to a basic artificial neural network that lacks a cycle in the connections between its nodes. The data flows in the direction of the output, from input to intermediate and intermediate to theoutcome layer.

Except for the central neuron (node), each layer in this network is directly linked. The ability of the Multilayer Feed-forward NN to categorize non-linear input through back propagation is one of its advantages. Error functions for every input were computed by the training algorithm rule, which sent the errors from one layer to one before it in each link that was weighted by the learning process that came before it.

4. RESULT EVALUATION

- **Classification Accuracy**
  Classification Accuracy is a statistical metric that assesses in NIDS accurate the forecasts the relative classification accuracy of the different proposed and present techniques is shown in Figure 5 and Table 1. Our recommended MDSO-EFNN method has a much better score of 95% when compared to Support Vector Machine (SVM) (80%), Back Propagation (BP) (84%), and K-Nearest Neighbors (KNN) (88%).
- **Precision Rate**
  Precision is defined as the proportion of cases that are classified to all instances of data that are accurately positive. The relativeprecision rate of the different proposed and present techniques is shown in Figure 6 and Table 2. Our recommended MDSO-EFNN method has a much better score of 97%, when compared to SVM (81.2%), BP (85.4%), and KNN (88.1%).
- **Sensitivity**
  Parameter sensitivity indicates to changing one or more network parameters affect the probability of a variable. The relative sensitivity of the different proposed and present techniques is shown in Figure 7 and Table 3. Our recommended MDSO-EFNN method has a much better score of 98%, when compared to SVM (85%), BP (88%), and KNN (84%)
- **Specificity**
  Specificity involves rule patterns, signatures and anomaly detection criteria to recognize and classify particular network threats. Specificity improves threat detection and reduces false positives. The relative sensitivity of the different proposed and present techniques is shown in Figure 8 and Table 4. Our recommended MDSO-EFNN method has a much better score of 98%, when compared to SVM (85%), BP (88%), and KNN (84%).

5. CONCLUSION

In the research, securing the CC concept is crucial for its success. We discuss several intrusions affecting cloud environment, it explained cloud NIDS kinds environment. Details on various intrusions detecting methods are included. The main drawback of NIDS is the need for sufficient time and training to effective. To overcome this problem, we proposed a MDSO-EFNN in the cloud environment to have presented a solution in the form of the security concerns in cloud computing are critical. Our method is centered on the analysis of network traffic flows in cloud computing environments. We use the NSL-KDD network traffic dataset to preprocess the data using Z-score normalization and extract pertinent information using the CWT and then to perform feature selection using ACO. The relative accuracy of the different proposed and present techniques is shown in figures and tables shows that, our recommended MDSO-EFNN method has a much better sensitivity score of 98.00% when compared to SVM, BP, KNN. The result was positive, and we determined that this work to be explored in the future with numerous more improvements and performance evaluations. Research will focus on selecting other ideal network features and the use of additional evolutionary ap-

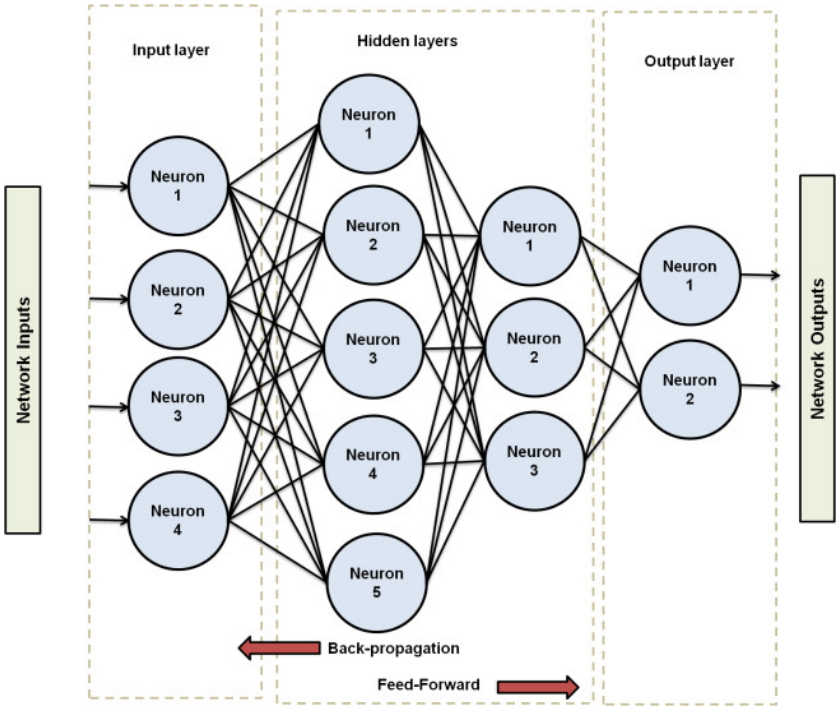

Fig. 4. Feed forward neural network

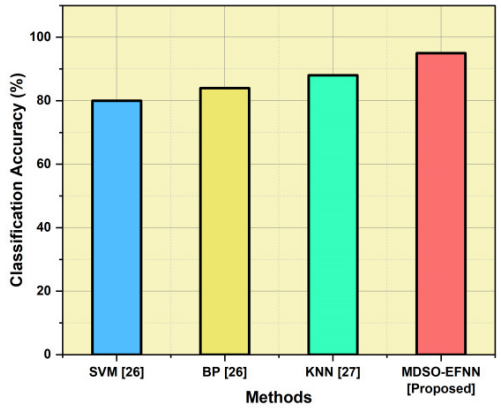

Fig. 5. Result of classification accuracy

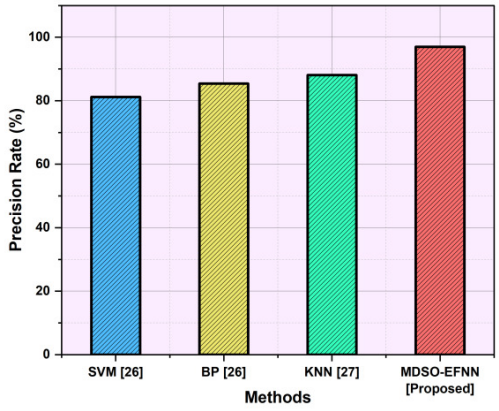

Fig. 6. Result of Precision rate

Table 1. Numerical outcomes of classification accuracy

Methods	Classification Accuracy (%)
SVM[26]	80
BP[26]	84
KNN[27]	88
MDSO-EFNN [Proposed]	95

Table 2. Numerical outcomes of Precision rate

Methods	Precision Rate (%)
SVM[26]	81.2
BP[26]	85.4
KNN[27]	88.1
MDSO-EFNN [Proposed]	97

proaches to optimize the control parameters of the classification algorithm.

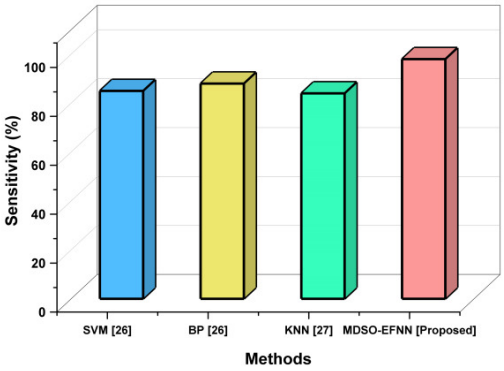

Fig. 7. Result of sensitivity

Table 3. Numerical outcomes of sensitivity

Methods	Sensitivity (%)
SVM[26]	85
BP[26]	88
KNN[27]	84
MDSO-EFNN [Proposed]	98

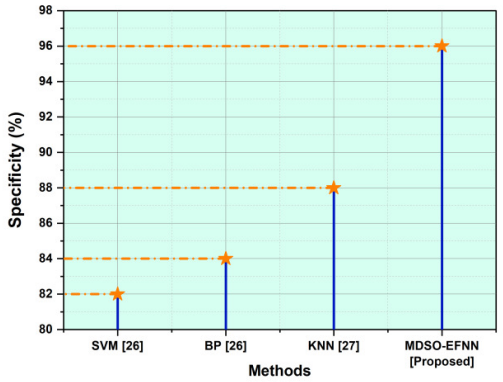

Fig. 8. Result of specificity

Table 4. Numerical outcomes of specificity

Methods	Specificity (%)
SVM[26]	82
BP[26]	84
KNN[27]	88
MDSO-EFNN [Proposed]	96

This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CCBY-4.0). View this license's legal deed at http://creativecommons.org/licenses/by/4.0 and legal code at http://creativecommons.org/licenses/by/4.0/legalcode for more information.

REFERENCES

1. Ud Din MM, Alshammari N, Alanazi SA, et al. InteliRank: A Four-Pronged Agent for the Intelligent Ranking of Cloud Services Based on End-Users' Feedback. *Sensors*. 2022;22(12):4627. doi:10.3390/s22124627

2. Dixit S, Hussain G. An effective intrusion detection system in cloud computing environment. In: *Mobile Radio Communications and 5G Networks: Proceedings of Third MRCN 2022*. Springer Nature Singapore; 2023:671-680. doi:10.1007/978-981-19-7982-8_56

3. Ali A, Alourani A. An Investigation of Cloud Computing and E-Learning for Educational Advancement. *International Journal of Computer Science & Network Security*. 2021;21(11):216-222.

4. Bingu R, Jothilakshmi S. Design of Intrusion Detection System using Ensemble Learning Technique in Cloud Computing Environment. *International Journal of Advanced Computer Science and Applications*. 2023;14(5). doi:10.14569/ijacsa.2023.0140580

5. Samunnisa K, Kumar GSV, Madhavi K. Intrusion detection system in distributed cloud computing: Hybrid clustering and classification methods. *Measurement: Sensors*. 2023;25:100612. doi:10.1016/j.measen.2022.100612

6. Thombre SU. *Network-Based Intrusion Detection System for Preventing the Cloud Computing Environment from Cyber-Attacks Using Deep Learning Algorithms*. Doctoral dissertation. Dublin, National College of Ireland; 2022.

7. You L, Wang Z. A Cloud Based Network Intrusion Detection System. *Tehničkivjesnik*. 2022;29(3):987-992. doi:10.17559/tv-20211130024245

8. Mayuranathan M, Saravanan SK, Muthusenthil B, Samydurai A. An efficient optimal security system for intrusion detection in cloud computing environment using hybrid deep learning technique. *Advances in Engineering Software*. 2022;173:103236. doi:10.1016/j.advengsoft.2022.103236

9. Jelidi M, Ghourabi A, Gasmi K. A hybrid intrusion detection system for cloud computing environments. In: *2019 International Conference on Computer and Information Sciences (ICCIS)*. IEEE; 2019:1-6. doi:10.1109/iccisci.2019.8716422

10. Chen L, Xian M, Liu J, Wang H. Intrusion detection system in cloud computing environment. In: *2020 International Conference on Computer Communication and Network Security (CCNS)*. IEEE; 2020:131-135. doi:10.1109/ccns50731.2020.00037

11. Selvapandian D, Santhosh R. Deep learning approach for intrusion detection in IoT-multi cloud environment. *Autom Softw Eng*. 2021;28(2):1-17. doi:10.1007/s10515-021-00298-7

12. Chen L, Xian M, Liu J, Wang H. Intrusion detection system in cloud computing environment. In: *2020 International Conference on Computer Communication and Network Security (CCNS)*. IEEE; 2020:131-135. doi:10.1109/ccns50731.2020.00037

13. Balamurugan E, Mehbodniya A, Kariri E, Yadav K, Kumar A, Anul Haq M. Network optimization using defender system in cloud computing security based intrusion detection system withgame theory deep neural network (IDSGT-DNN). *Pattern Recognition Letters*. 2022;156:142-151. doi:10.1016/j.patrec.2022.02.013

14. Sathiyadhas SS, Soosai Antony MCV. A network intrusion detection system in cloud computing environment using dragonfly improved invasive weed optimization integrated Shepard convolutional neural network. *International Journal of Adaptive Control and Signal Processing*. 2022;36(5):1060-1076. doi:10.1002/acs.3386

15. Alheeti KMA, Lateef AAA, Alzahrani A, Imran A, Al_Dosary D. Cloud Intrusion Detection System Based on SVM. *Int J Interact Mob Technol*. 2023;17(11):101-114. doi:10.3991/ijim.v17i11.39063

16. Alghamdi R, Bellaiche M. A deep intrusion detection system in lambda architecture based on edge cloud computing for IoT. In: *2021 4th International Conference on Artificial Intelligence and Big Data (ICAIBD)*. IEEE; 2021:561-566. doi:10.1109/icaibd51990.2021.9458974

17. Aldallal A, Alisa F. Effective intrusion detection system to secure data in cloud using machine learning. *Symmetry*. 2021;13(12):2306. doi:10.3390/sym13122306

18. Shukla AK, Sharma A. Cloud Base Intrusion Detection System using Convolutional and Supervised Machine Learning. In: *2023 6th International Conference on Information Systems and Computer Networks (ISCON)*. IEEE; 2023:1-5. doi:10.1109/iscon57294.2023.10112007

19. Maheswari KG, Siva C, Priya GN. An Optimal Cluster Based Intrusion Detection System for Defence Against Attack in Web and Cloud Computing Environments. *Wireless Pers Commun*. 2023;128(3):2011-2037. doi:10.1007/s11277-022-10030-7

20. Sharon A, Mohanraj P, Abraham TE, Sundan B, Thangasamy A. An intelligent intrusion detection system using hybrid deep learning approaches in cloud environment. In: *International Conference on Computer, Communication, and Signal Processing*. Springer International Publishing; 2022:281-298. doi:10.1007/978-3-031-11633-9_20

21. Alduailij M, Khan QW, Tahir M, Sardaraz M, Alduailij M, Malik F. Machine-learning-based DDoS attack detection using mutual information and random forest feature importance method. *Symmetry*. 2022;14(6):1095. doi:10.3390/sym14061095

22. Shyla SI, Sujatha SS. Cloud security: LKM and optimal fuzzy system for intrusion detection in cloud environment. *Journal of Intelligent Systems*. 2019;29(1):1626-1642. doi:10.1515/jisys-2018-0479

23. Bibi A, Sampedro GA, Almadhor A, Javed AR, Kim T hoon. A Hypertuned Lightweight and Scalable LSTM Model for Hybrid Network Intrusion Detection. *Technologies*. 2023;11(5):121. doi:10.3390/technologies11050121

24. Wang J, Wang L. SDN-Defend: A Lightweight Online Attack Detection and Mitigation System for DDoS Attacks in SDN. *Sensors*. 2022;22(21):8287. doi:10.3390/s22218287

25. Jaber AN, Rehman SU. FCM–SVM based intrusion detection system for cloud computing environment. *Cluster Comput*. 2020;23(4):3221-3231. doi:10.1007/s10586-020-03082-6

26. Xiao Y, Xiao X. An intrusion detection system based on a simplified residual network. *Information*. 2019;10(11):356. doi:10.3390/info10110356

27. Wazirali R. An improved intrusion detection system based on KNN hyperparameter tuning and cross-validation. *Arab J Sci Eng*. 2020;45(12):10859-10873. doi:10.1007/s13369-020-04907-7