Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Optimization of a Deep Learning-Based Model for Detecting DDoS Attacks in Cloud Computing

Dr. Abhijeet Madhukar Haval[1], Sushree Sasmita Dash[2]

[1]Associate Professor, Faculty of CS & IT, Kalinga University, Naya Raipur, Chhattisgarh, India, ku.abhijeetmadhukarhaval@kalingauniversity.ac.in
[2]Assistant Professor, Faculty of CS & IT, Kalinga University, Naya Raipur, Chhattisgarh, India, ku.sushreesasmitadash@kalingauniversity.ac.in

Intrusion Detection Systems (IDS) in the Cloud Computing (CC) environment have garnered significant attention in recent years. Deep Learning (DL)-based Intrusion Detection Systems (IDS) are the most effective ways of detecting threats. Distributed Denial of Service (DDoS) assaults are prevalent in the CC environment. Authorized end-users may experience unavailability of CC services due to excessive network activity, leading to loss of money. DL approaches are suitable and efficient algorithms for classifying standard and malicious data in this scenario. Therefore, this paper proposes a unique approach called Ensemble Feature Selection-Deep Neural Network (EFS-DNN) to detect DDoS attacks effectively. The provided dataset undergoes a preliminary pre-processing phase, in which the min-max normalization approach is used to substitute all input values inside a given range. Subsequently, the standardized data is inputted into the suggested EFS to identify the most suitable collection of features, thereby facilitating the procedure for classification. The EFS employs mutual grouping among Particle Swarm Optimizer (PSO), Grey Wolf Optimizer (GWO), and Whale Optimization Algorithm (WOA) to identify the optimal features that significantly impact the detection efficiency of DDoS attacks in CC. The chosen characteristics are submitted to a DNN classifier to classify normal and malicious data. The method will be implemented using the MATLAB software and evaluated via experimentation, demonstrating a detection accuracy of 96.12% for DDoS attacks.

Keywords: Deep learning, Intrusion Detection System, DDoS attacks, Cloud Computing, Ensemble Feature Selection, Deep Neural Network, PSO, GWO, WOA.

1. Introduction

CC enables consumers and companies to save infrastructure expenses by offering a range of online tools. These materials are offered as services. Users and businesses are charged based on the length of service consumption, following a pay-as-you-use philosophy [1]. The accessibility of these resources is crucial since any disruption may result in significant financial and reputational damage for users/organizations. Malicious actors can use DDoS operations to render certain CC services inaccessible to authorized users. In this assault, the assailants exerted an exceedingly heavy burden on the services supplied by a target server on the public network. This leads to the complete use of the perpetrator's bandwidth, rendering it inaccessible [24][26].

This sort of attack utilizes a network known as a botnet, which consists of several hacked computers on the Internet [30][32]. The purpose of the assault is to direct traffic towards the victim. Reflection and magnification methods are used to enhance the devastating impact of this assault. During reflection-based assaults, the bots do not immediately transmit data to the target. Instead, the data is redirected to accessible servers, known as reflectors, by sending requests with a falsified source IP address that seems to be the victim's [2][25]. In response to these demands, the servers generate significant traffic that overwhelms the victim's capacity, causing it to become exhausted. The victim receives answers of much greater magnitude than the bots' queries, a phenomenon known as amplification. The effect of this kind of attack is quantified using an amplification aspect, which is determined by comparing the size of the answer to the size of the query. An instance of this kind of assault is the DNS amplification attack [3].

In this scenario, the bots initiate requests to activate recursive DNS resolvers. During these queries, the origin IP address is manipulated to seem like the IP address of the target, and certain parameters are provided to trigger a substantial reply. When replying to these inquiries, DNS resolvers generate a substantial amount of traffic towards the target's IP address. This leads to bandwidth depletion at the target location, eventually rendering it inaccessible. Multiple machine learning algorithms have been suggested to identify DDoS threats in CC. The primary obstacle with ML-based systems is achieving precise identification of these threats. Extreme Learning Machine (ELM) is classified as an Artificial Neural Network (ANN) with just one hidden layer, known as a Single Hidden Layer Feed-forward Neural Network SLFNN. In this approach, the input-hidden link weights, which represent the weights for connections between the input and concealed layers and the concealed biases, are configured with random values.

The Moore-Penrose inverse approach is used to compute the weights of links among the concealed and output layers, often known as hidden-output connection values [4][27]. Thus, it may be taught in a solitary iteration and does not need repeated training. Occasionally, the random assignment of bias and weights during initialization may lead to suboptimal training precision, decreasing testing precision. Additionally, the concealed layer of the SLFNN should have more neurons than other layers, leading to increased testing duration for unfamiliar samples. The research suggests that optimizing the input-hidden connection weights and concealed biases for ELM is recommended to address these issues. Different genetic optimization strategies are used to optimize these parameters.

Cao et al. (2012) introduce a more advanced iteration of E-ELM known as the Self-adaptive Evolutionary Extreme Learning Machine (SaE-ELM) [5]. SaE-ELM has integrated several advanced functionalities to enhance the efficiency of determining the best values of ELM parameters, such as input-hidden connection weights and concealed biases. During every generation, the system may autonomously select the most appropriate mutation approach for an individual in the community. Each individual is assigned a randomly chosen scaling factor from a predetermined range throughout the mutation process. The crossover rate is dynamically adjusted during the whole evolutionary process. Nevertheless, this model utilizes just one crossover administrator, the uniform operator. Integrating numerous crossover operators helps enhance the search process for determining the best values of ELM parameters[6][29].

During DDoS assaults, several attackers manipulate false data detection over a wide spectrum, posing an obstacle for the victim and their immediate network connections. This kind of attack is launched by taking advantage of vulnerabilities in the system, causing a significant amount of failed network traffic to overload certain resources such as memory, network processing time, and bandwidth [7]. This results in severe interruption to the victim. This assault can be conducted from many sources, where several hosts collaborate to launch attack packages against the target. Alternatively, a specific source attack may originate from a single host. Presently, assault toolboxes are designed to be easily accessible on the Internet [8] [31]to guarantee that any Internet users may use these kits to initiate assaults with little complications. Consequently, there has been an increase in the number of tests due to the creation of a system designed to combat DDoS assaults [9][10].

The main findings of this study include:

- Enhancing the current body of knowledge by introducing a very effective IDS for identifying cyber-attacks in CC environments.

- Suggesting the use of an EFS method to identify the most important subset of characteristics that would improve the effectiveness of detecting DDoS assaults.

- A novel method named Ensemble Feature Selection-Deep Neural Network (EFS-DNN) detection has been introduced to mitigate DDoS assaults efficiently.

2. Literature survey

The following is an evaluation of literature reviews conducted by different researchers on DDoS detection in CC. Velliangiri et al. [33] introduced a TEHO-enabled DBN, which was used to detect assaults at early stages. However, this approach involves a higher number of iterations. It utilizes TEHO-DBN to update the weights of the input and hidden units in the MLP layer, which results in increased computing time. Authors in [11] created SD-LVQ to circumvent this limitation. They investigated using cloud-mounted computers to minimize detection tactics of DDoS-encrypted cross-site attacks. Nevertheless, the deep-supervised algorithms still pose a hurdle for the hybrid cloud data center [34]. The challenge in [11] was eliminated in [12][28].

Doriguzzi-Corin et al. [12] developed the LUCID model architecture, prioritizing a lightweight application with little processing overhead and quick detection time. However, the approach exhibited limited convergence and accuracy. The issue of low convergence was resolved in [13]. Authors in [13] created FS-WOA, effectively preventing DDoS attacks from infiltrating large-scale industries. Nevertheless, this strategy fails to generate distinct instances to identify new assaults.

Authors in [14] introduced the SaE-ELM-Ca model. Despite being created to discover the optimal number of hidden neurons to enhance the learning potential of the model, this strategy did not effectively leverage numerous connections for testing and instead relied on a single connection. Alduailij et al. [15] introduced the use of Mutual Information (MI) and Random Fourier Features (RFF) as a means to decrease misclassification errors via the use of several classifiers. Nevertheless, this approach was ineffective in analyzing with DL-based detection, which was subsequently improved in [16]. Alqarni [16] proposed using an ensemble technique for detecting DDoS attacks, effectively controlling the size of the feature and dataset, resulting in improved performance. During its implementation, prevailing difficulties resulted in a longer duration.

Authors in [17] imposed restrictions on using time and designing a feed forward-based DNN for DDoS attack detection. This strategy achieved precise and rapid outcomes in a condensed timeframe. However, this strategy prioritized the mandatory training procedure due to including many packages in the dataset, which was not favored in other current methods. Bovenzi et al. [18] used the Multi-Modal Deep AutoEncoder (M2-DAE) model to detect intrusions in IoT [36]. This strategy was designed to maintain privacy and use distributed approaches to achieve high effectiveness and adaptability.

Nevertheless, the assessment of attack classes was not conducted using this methodology. An ML technique was used by Ahmad et al. [19] to categorize network threats. Here, a comprehensive range of sophisticated characteristics was considered for the first categorization. This strategy achieved a high F-measure but did not consider additional datasets.

Table 1: Summary of literature survey

Author(s)	Method Used	Advantages	Disadvantages
Velliangiri, S., Karthikeyan, P., & Vinoth Kumar, V. (2021)	Optimization-based deep networks	High accuracy in detecting DDoS attacks	Computational complexity
Arul, E., & Punidha, A. (2021)	Supervised deep learning vector quantization	Effective in detecting MemCached DDoS malware attacks	Require large labeled datasets for training
Doriguzzi-Corin, R., Millar, S., Scott-Hayward, S., Martinez-del-Rincon, J., & Siracusa, D. (2020)	Lightweight deep learning (LUCID)	Practical and lightweight solution, efficient for real-time detection	Limited to specific types of DDoS attacks
Agarwal, A., Khari, M., & Singh, R. (2022)	Deep learning model in cloud storage application	Robust detection mechanism, high accuracy	May not generalize well to all cloud environments
Kushwah, G. S., & Ranga, V. (2021)	Optimized extreme learning machine	Fast training and detection, suitable for high-speed networks	Might be less effective with evolving attack strategies
Alduailij, M., Khan, Q. W., Tahir, M., Sardaraz, M., Alduailij, M., & Malik, F. (2022)	Mutual information and random forest feature importance	High detection accuracy, effective feature selection method	Computationally intensive, especially for large datasets
Alqarni, A. A. (2022)	Majority vote-based	Increased detection	Complexity in integrating

Author(s)	Method Used	Advantages	Disadvantages
	ensemble approach	accuracy through ensemble methods	multiple models and managing ensemble
Cil, A. E., Yildiz, K., & Buldu, A. (2021)	Feed forward based deep neural network model	High accuracy and efficiency in detecting DDoS attacks	May require significant computational resources for training
Bovenzi, G., Aceto, G., Ciuonzo, D., Persico, V., & Pescapé, A. (2020)	Hierarchical hybrid intrusion detection approach in IoT scenarios	Effective for IoT scenarios, combining multiple detection methods	Complexity in implementation and maintenance
Ahmad, Z., Shahid Khan, A., Wai Shiang, C., Abdullah, J., & Ahmad, F. (2021)	Systematic study of machine learning and deep learning approaches	Comprehensive overview, identifies strengths and weaknesses of various approaches	Generalized study, may not provide specific actionable insights for particular applications

Table 1 summarizes the methods, advantages, and disadvantages of different approaches to detecting DDoS attacks and network intrusions as presented in the referenced works.

3. Proposed methodology

A DDoS attack involves overwhelming an attacker's network components with many fake assaulting packets originating from several computers.

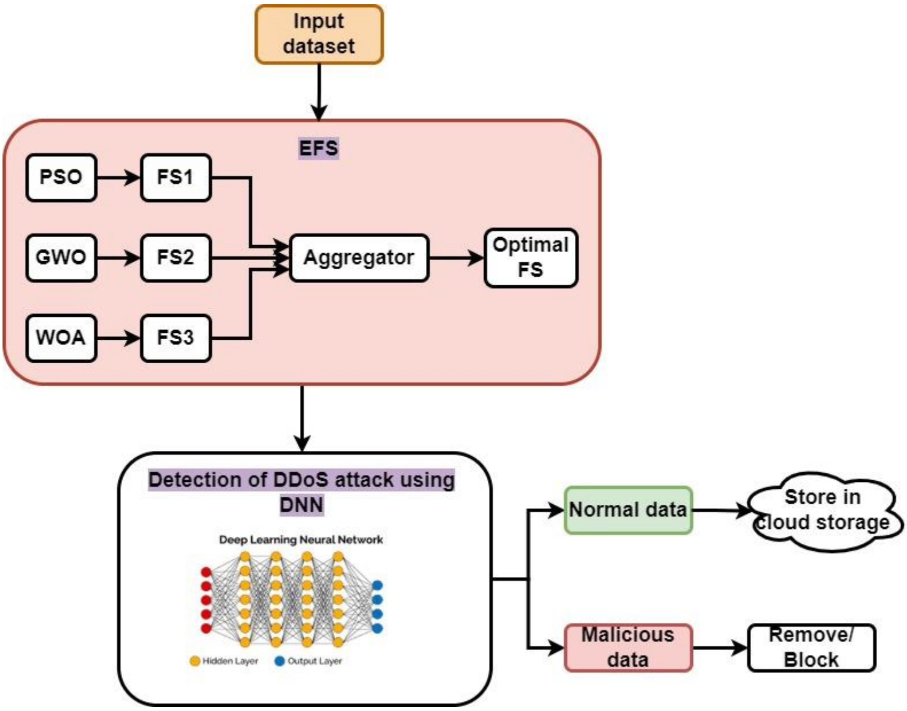

Fig. 1 Architecture of the proposed Ensemble Feature Selection-Deep Neural Network (EFS-DNN) to effectively detect DDoS attacks

An effective assault grants the attacker entry into the victim's workstation, enabling the theft of valuable internal information and perhaps causing interruption and service delays in some instances. Therefore, it is essential to develop an IDS-based safety system that assists administrators in proactively notifying or alerting them in the event of any harmful actions or violations of security regulations. A minimal IDS utilizes DL techniques to classify normal and malicious information. The primary benefit of researching intrusion prevention models, namely in the case of DDoS attacks, is their significant impact on large-scale international corporations. Furthermore, the suggested technique is suited for IDS and relevant to the safety model in a cloud context. The suggested approach ensures that private data is safeguarded and immune to subsequent attacks via secure storage. The suggested architecture is seen in Fig. 1 below.

The suggested framework's architecture is shown in Fig. 1, illustrating the proposed Ensemble Feature Selection-Deep Neural Network (EFS-DNN) to detect DDoS attacks effectively. The dataset is first pre-processed using the Min-Max standardization approach to normalize the information within a given range. The subsequent phase involves selecting pertinent and suitable features using the provided EFS technique. The chosen features are analyzed using an efficient DNN classification model to distinguish between normal and malicious data.

a. Pre-processing

The input dataset is initially created and then subjected to a normalizing process. This technique converts the characteristics with larger quantities, which have a significant impact, into limited numeric values. Min-max standardization is performed using linear transformation to transfer the value of an attribute ranging from $[\mathrm{min}(q), \mathrm{max}(q)]$ to $[\mathrm{newmin}(q), \mathrm{newmax}(q)]$.

b. Ensemble Feature Selection

An EFS strategy is intended to address the restrictions imposed by using a single FS algorithm. This approach utilizes bilateral information to identify the most optimum collection of features. The three main FS algorithms used are PSO [20], GWO [21], and WOA [22][35]. The pre-processed database is concurrently inputted into various algorithms, with each algorithm generating the optimal FS subgroup. Thus, this phase generates four distinct sets of features.

The subsections are inputted into the aggregator to integrate the selected group of features based on the common information between features and classes. The individual responsible for combining obtains the initial characteristics from the selected subgroups. The top-ranked related features acquire the common feature as an optimal subgroup without calculating the shared data between features and the data between features and classes. However, the differentiating features compute the shared data between FS and category for each feature, considering the most significant shared data. This feature subgroup computes the shared data between features using the selected characteristics as the optimal choice. If the shared information between the features picked is lower than the customized threshold α, the feature will be chosen. The FS exchanged data is used to quantify the importance of a random characteristic concerning the selected attributes. Based on extensive study, the threshold value is determined to be $\alpha = 0.75$. In the ensemble technique, an 'aggregator' plays a vital role in combining several FS algorithms. The person accountable for integrating the proposed approach of choosing EFS emphasizes the decrease in redundancy within the selected group of attributes by including feature class and shared data. This approach also prevents the bias in FS that arises from relying solely on a single FS algorithm.

c. Ensemble Feature Selection-Deep Neural Network (EFS-DNN) to detect DDoS attacks

The chosen characteristics obtained from EFS are inputted into the input nodes of the DNN framework. The DNN architecture facilitates the classification of both regular and attacked packets of information. Essentially, DNN is a kind of neural network identical to a regular Neural Network (NN), with the main distinction being that DNNs include numerous concealed layers between the input and output layers. The DNN architecture consists of two distinct phases: training and testing. This DL approach is particularly efficient when a bigger dataset is used during training. The input values are multiplied by their respective weights and then summed along with the neuron's bias in the concealed layer. This process is formally expressed by Equation 1:

$$C_{h(x)} = \left(\sum_{m=1}^{M} w_{xm} F_{I_j}\right) + b_x \tag{1}$$

The variable b_x represents the bias, which has a constant value. The value of x is set to 1, 2, and so on. Let K be the number of input and concealed nodes. The link weight between the input and concealed layer is represented by w_{xm}. The variables M and K represent the input and concealed neurons in the first concealed layer, respectively. F_{I_j} represents the chosen set of optimum features produced via EFS. This set is used as input for the DL strategy, where m is a value between 1 and M. $C_{h(x)}$ represents the output of the concealed layer of the whole network. The outcome of the concealed layer is determined by the activation function, as defined in Equation 2:

$$A\left(C_{h(x)}\right) = \frac{1}{1+e^{-C_{h(x)}}} \tag{2}$$

The sigmoid activation function is denoted by the symbol " $A(.)$". Eq. 3 expresses the mathematical calculation at the output layer, where the output of the concealed layer is multiplied by the weights connecting the hidden and output layers. This product is then added to the bias b_x function.

$$C_{O(x)} = A\left(\sum_{n=1}^{N} w_{xn} A\left(C_{h(x)}\right)\right) + b_x \tag{3}$$

The variable w_{xn} represents the weight that connects the concealed and output layers. The output layer's activation function serves as the complete network's final output. The ultimate result is chosen by decreasing the error function achieved in the system model. The evaluation is performed by computing the Mean Square Error (MSE), characterized as the discrepancy between the true and anticipated likelihood values, as shown in Equation 4.

$$MSE = \frac{1}{T}\sum_{t=1}^{T}\left(C_{true,T} - C_{anticipated,T}\right)^2 \tag{4}$$

The terms "$C_{true,T}$" and "$C_{anticipated,T}$"refers to the classification outcomes of true and anticipated classes for normal and malicious data. The error value is reduced with each repetition during the training phase. Once the network has completed its training, the system can categorize data based on the topology of the trained network. Once the error function reaches its minimal value, the network will halt the iteration and safeguard private data in cloud storage.

Therefore, using the suggested approach, both the detection and false rates are concurrently enhanced. This, in turn, effectively identifies DDoS attacks and provides a high degree of safety in cloud storage. Therefore, it can be inferred that the general efficiency would be improved by implementing the suggested EFS-DNN model for detecting DDoS attacks.

4. Results and discussion

This section presents the empirical findings of a DDoS IDS employing the EFS-DNN approach. The execution is performed using the MATLAB software, running on an Intel (R) Core (TM) i5-3570S CPU processor with a clock rate of 3.10 GHz. Several assessment metrics such as detection Accuracy, Sensitivity, Specificity, and Error values are assessed and tested to validate the efficiency of the suggested system. The current methods used for comparison include Support Vector Machine (SVM), K-Nearest Neighbour (KNN), Artificial Neural Network (ANN), and the newly suggested EFS-DNN. In this scenario, 75% of the dataset is allocated for the training stage, while the remaining 25% of images are reserved for the testing stage. The case study utilizes the publicly accessible CIC-IDS 2017 database [23]. The CIC-IDS2017 database consists of both benign and mostly contemporary assaults that are prevalent in society, providing an accurate representation of data from the real world.

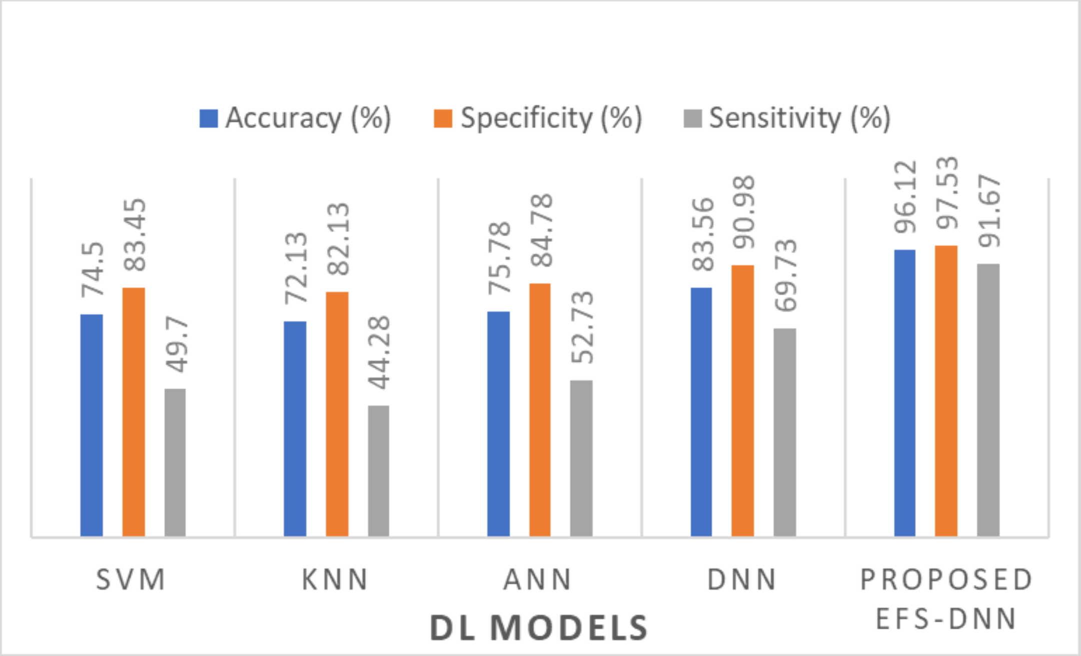

Fig. 2 Performance comparison (in %) of various DL models to detect DDoS attack in CC environment.

Fig. 2 comprehensively examines the performance metrics for several DL models, including SVM, KNN, ANN, DNN, and the Proposed EFS-DNN. The corresponding percentages for accuracy, specificity, and sensitivity accompany each model. The EFS-DNN model exhibits superior performance in all measures, achieving an accuracy of 96.12%, specificity of 97.53%, and sensitivity of 91.67%. This demonstrates its better efficacy in detecting DDoS assaults than other models. The DNN model has significant performance, with an accuracy of 83.56%, specificity of 90.98%, and sensitivity of 69.73%. The ANN, SVM, and KNN models exhibit somewhat worse performance, with SVM and KNN showing the lowest sensitivity rates of 49.7% and 44.28%, respectively. The comparison demonstrates the efficacy of the proposed EFS-DNN model in consistently and adequately identifying DDoS assaults.

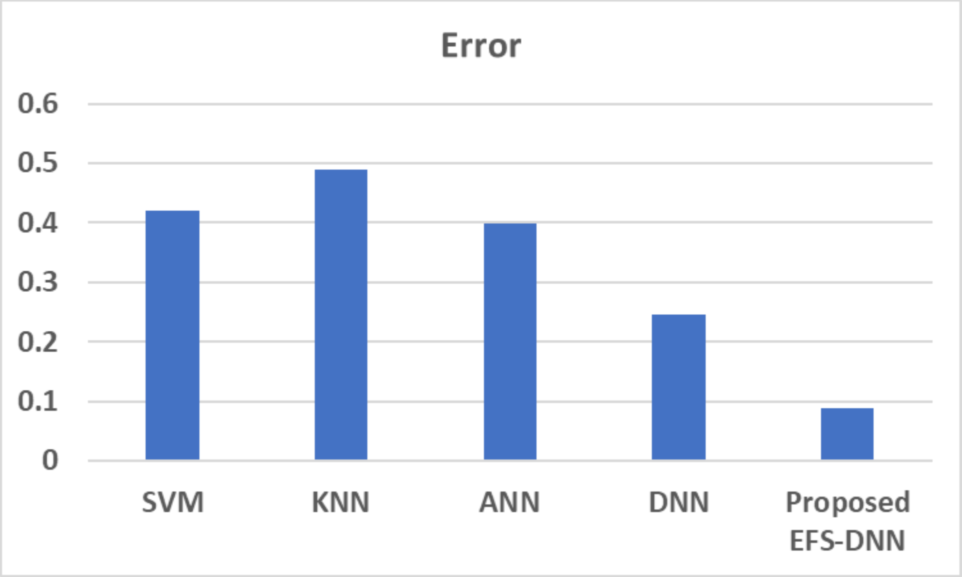

Fig. 3 Error values of various DL models to detect DDoS attacks in CC environment.

Fig. 3 displays the performance of several DL models based on their error levels. The EFS-DNN model, as described, demonstrates exceptional performance with a minimal error value of 0.0874. This result signifies the model's superior accuracy and dependability in effectively identifying DDoS assaults in a CC setting. The performance of this model is much better than the other models: DNN with an error rate of 0.2456, ANN with 0.3987, SVM with 0.4195, and KNN with 0.4897. The reduced error value of EFS-DNN indicates its improved efficacy in precisely detecting DDoS assaults in comparison to conventional and more advanced models, which have larger error rates and, thus, lesser accuracy.

5. Concluding remarks

This research introduces a novel method called Ensemble Feature Selection-Deep Neural Network (EFS-DNN) for accurately identifying DDoS assaults. The submitted dataset undergoes a pre-processing step when the min-max normalization strategy replaces all input values inside a specified range. Afterward, the standardized data is entered into the recommended EFS to determine the optimal set of features, making the classification process easier. The EFS utilizes a bilateral grouping of PSO, GWO, and WOA to determine the best characteristics that substantially influence the detection efficiency of DDoS assaults in CC. The selected attributes are inputted into a DNN classifier to categorize normal and malicious data. The method will be implemented using the MATLAB software and evaluated via experimentation. The EFS-DNN model demonstrates the highest detection accuracy of 96.12% and the lowest error value of 0.0874 for DDoS attacks.

References

1. Kishor, K. (2023). Impact of cloud computing on entrepreneurship, cost, and security. In Cloud-based Intelligent Informative Engineering for Society 5.0 (pp. 171-191). Chapman and Hall/CRC.
2. Bobir, A.O., Askariy, M., Otabek, Y.Y., Nodir, R.K., Rakhima, A., Zukhra, Z.Y., Sherzod, A.A. (2024). Utilizing Deep Learning and the Internet of Things to Monitor the Health of Aquatic Ecosystems to Conserve Biodiversity. Natural and Engineering Sciences, 9(1), 72-83.
3. Rajendran, B. (2020, February). DNS amplification & DNS tunneling attacks simulation, detection and mitigation approaches. In 2020 International Conference on Inventive Computation Technologies (ICICT) (pp. 230-236). IEEE.
4. Trivedi, J., Devi, M. S., & Solanki, B. (2023). Step Towards Intelligent Transportation System with Vehicle Classification and Recognition Using Speeded-up Robust Features. Archives for Technical Sciences, 1(28), 39-56.
5. Bacanin, N., Stoean, C., Zivkovic, M., Jovanovic, D., Antonijevic, M., & Mladenovic, D. (2022). Multi-swarm algorithm for extreme learning machine optimization. Sensors, 22(11), 4204.
6. Ho, S.M., & Lee, H. (2012). A Thief among Us: The Use of Finite-State Machines to Dissect Insider Threat in Cloud Communications. Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, 3(1/2), 82-98.
7. Aamir, M., & Zaidi, S. M. A. (2021). Clustering based semi-supervised machine learning for DDoS attack classification. Journal of King Saud University-Computer and Information Sciences, 33(4), 436-446.
8. Malathi, K., Anandan, R., & Vijay, J. F. (2023). Cloud Environment Task Scheduling Optimization of Modified Genetic Algorithm. Journal of Internet Services and Information Security, 13(1), 34-43.
9. Annamalai, S., Udendhran, R., & Vimal, S. (2019). An intelligent grid network based on cloud computing infrastructures. In Novel practices and trends in grid and cloud computing (pp. 59-73). IGI Global.
10. Kumar, A., Joshi, P., Bala, A., Sudhakar Patil, P., Jang Bahadur Saini, D. K., & Joshi, K. (2023). Smart Transaction through an ATM Machine using Face Recognition. Indian Journal of Information Sources and Services, 13(2), 7–13.
11. Arul, E., & Punidha, A. (2021). Supervised deep learning vector quantization to detect MemCached DDOS malware attack on cloud. SN Computer Science, 2(2), 85.
12. Doriguzzi-Corin, R., Millar, S., Scott-Hayward, S., Martinez-del-Rincon, J., & Siracusa, D. (2020). LUCID: A practical, lightweight deep learning solution for DDoS attack detection. IEEE Transactions on Network and Service Management, 17(2), 876-889.
13. Agarwal, A., Khari, M., & Singh, R. (2022). Detection of DDOS attack using deep learning model in cloud storage application. Wireless Personal Communications, 1-21.

14. Kushwah, G. S., & Ranga, V. (2021). Optimized extreme learning machine for detecting DDoS attacks in cloud computing. Computers & Security, 105, 102260.
15. Alduailij, M., Khan, Q. W., Tahir, M., Sardaraz, M., Alduailij, M., & Malik, F. (2022). Machine-learning-based DDoS attack detection using mutual information and random forest feature importance method. Symmetry, 14(6), 1095.
16. Alqarni, A. A. (2022). Majority vote-based ensemble approach for distributed denial of service attack detection in cloud computing. Journal of Cyber Security and Mobility, 265-278.
17. Cil, A. E., Yildiz, K., & Buldu, A. (2021). Detection of DDoS attacks with feed forward based deep neural network model. Expert Systems with Applications, 169, 114520.
18. Bovenzi, G., Aceto, G., Ciuonzo, D., Persico, V., & Pescapé, A. (2020, December). A hierarchical hybrid intrusion detection approach in IoT scenarios. In GLOBECOM 2020-2020 IEEE global communications conference (pp. 1-7). IEEE.
19. Ahmad, Z., Shahid Khan, A., Wai Shiang, C., Abdullah, J., & Ahmad, F. (2021). Network intrusion detection system: A systematic study of machine learning and deep learning approaches. Transactions on Emerging Telecommunications Technologies, 32(1), e4150.
20. Gad, A. G. (2022). Particle swarm optimization algorithm and its applications: a systematic review. Archives of computational methods in engineering, 29(5), 2531-2561.
21. Nadimi-Shahraki, M. H., Taghian, S., & Mirjalili, S. (2021). An improved grey wolf optimizer for solving engineering problems. Expert Systems with Applications, 166, 113917.
22. Juma, J., Mdodo, R.M., & Gichoya, D. (2023). Multiplier Design using Machine Learning Alogorithms for Energy Efficiency. Journal of VLSI Circuits and Systems, 5(1), 28-34.
23. https://www.kaggle.com/datasets/chethuhn/network-intrusion-dataset
24. Kutlu, Y., & Camgözlü, Y. (2021). Detection of coronavirus disease (COVID-19) from X-ray images using deep convolutional neural networks. Natural and Engineering Sciences, 6(1), 60-74.
25. Arora, A., Yadav, S. K., & Sharma, K. (2021). Denial-of-service (dos) attack and botnet: Network analysis, research tactics, and mitigation. In Research Anthology on Combating Denial-of-Service Attacks (pp. 49-73). IGI Global.
26. Jelena, T., & Srđan, K. (2023). Smart Mining: Joint Model for Parametrization of Coal Excavation Process Based on Artificial Neural Networks. Archives for Technical Sciences, 2(29), 11-22.
27. Zhang, W., Wu, Q. J., Yang, Y., & Akilan, T. (2020). Multimodel feature reinforcement framework using Moore–Penrose inverse for big data analysis. IEEE Transactions on Neural Networks and Learning Systems, 32(11), 5008-5021.
28. Shiraishi, Y., Mohri, M., & Fukuta, Y. (2011). A Server-Aided Computation Protocol Revisited for Confidentiality of Cloud Service. Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, 2(2), 83-94.
29. Sanjalawe, Y., & Althobaiti, T. (2023). DDoS Attack Detection in Cloud Computing Based on Ensemble Feature Selection and Deep Learning. Computers, Materials & Continua, 75(2).
30. Gali, M., & Mahamkali, A. (2022). A Distributed Deep Meta Learning based Task Offloading Framework for Smart City Internet of Things with Edge-Cloud Computing. Journal of Internet Services and Information Security, 12(4), 224-237.
31. Vimal, S., Kalaivani, L., & Kaliappan, M. (2019). Collaborative approach on mitigating spectrum sensing data hijack attack and dynamic spectrum allocation based on CASG modeling in wireless cognitive radio networks. Cluster Computing, 22, 10491-10501.
32. Kumar, A., Joshi, P., Bala, A., Sudhakar Patil, P., Jang Bahadur Saini, D. K., & Joshi, K. (2023). Smart Transaction through an ATM Machine using Face Recognition. Indian Journal of Information Sources and Services, 13(2), 7–13.
33. Velliangiri, S., Karthikeyan, P., & Vinoth Kumar, V. (2021). Detection of distributed denial

of service attack in cloud computing using the optimization-based deep networks. Journal of Experimental & Theoretical Artificial Intelligence, 33(3), 405-424.

34. Jayasree, V., Nithya, M., & Prabaharan, S. (2012). Cloud Data Retrieval for Multi related keyword based on Clustering Technology. International Journal of Communication and Computer Technologies (IJCCTS), 1(1), 60-66.
35. Rana, N., Latiff, M. S. A., Abdulhamid, S. I. M., & Chiroma, H. (2020). Whale optimization algorithm: a systematic review of contemporary applications, modifications and developments. Neural Computing and Applications, 32, 16245-16277.
36. Alnumay, W. S. (2024). The Past and Future Trends in IoT Research. National Journal of Antennas and Propagation, 6(1), 13-22.