International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Anomaly based Intrusion Detection System using Hybrid ResNet50 and 3D Convolutional Neural Network

K. B. Teena*[1], Swati Sharma[2]

Submitted: 25/01/2024 **Revised**: 03/03/2024 **Accepted**: 11/03/2024

Abstract: An Intrusion Detection System (IDS) is used to monitor and analyze data to identify any intrusions that have been made into a system or network. Because of the large capacity of network data with redundant and irrelevant features, accurate and precise intrusion detection is challenging. To overcome this problem, this research proposed the Deep Learning (DL) approach of ResNet50 and 3D Convolutional Neural Network (CNN) with Black Widow Optimization (BWO) for detecting the network attacks in IDS. Initially, the input data is acquired from three public datasets named CICDDoS2019, NSL-KDD, and UNSW-NB15 datasets. The collected data is then pre-processed to preserve the redundancy using normalization. The pre-processed output is then provided to the feature selection process for compiling the optimum features by using Principal Component Analysis (PCA). The network weights are optimized using BWO to obtain the optimum solution for identifying intruders. According to the findings, the proposed method is more efficient and it achieves accuracy of 0.93, 0.96 and 0.99 on CICDDoS2019, UNSW-NB15 and NSL-KDD datasets when compared to existing methods like Deep Feed Forward Neural Network (DFNN), CNN and Generative Adversarial Network (GAN).

Keywords: *Black Widow Optimization, Convolutional Neural Network, Deep Learning, Intrusion Detection System, ResNet50.*

1. Introduction

There has been a substantial growth in the Internet's importance to modern life. In today's world, internet-based information processing systems are vulnerable to a wide range of attacks that may cause damage in different ways. Thus, the value of information security is rapidly increasing [1][2]. The primary objective of information security is to design foolproof information systems that cannot be penetrated, modified, or destroyed by outside forces. As a bonus, information security helps reduce threats to privacy, authenticity, and accessibility-the three pillars of information security [3][4]. Many anti-Internet-based attack technologies have been developed for efficient detection of intrusions over the years. IDS are the most crucial activities for security infrastructure in network environments and are majorly utilized for preventing threats [5][6]. Moreover, IDSs provide a barrier against the assault of online computer systems. In case when a standard firewall is ineffective, IDS might be used to monitor network traffic and user activity on computer systems for the identification of malicious activity [7][8]. The principle behind intrusion detection is the idea that unauthorized users' actions will be distinguished from those of authorized ones. It is possible to identify intrusions into any information system by using a method called "Intrusion Detection." Intrusions are actions taken against a system that breaches and violates its security policy [9][10]. One of the most significant kinds of security software that has been utilized to deal with intrusions is known as an IDS. The data security system must have many different types of systems, one of which is an IDS [11][12]. IDS is utilized for the detection, monitoring, and analysis of the susceptibilities of hardware and software running in a network [13][14]. IDS can also look for any violation of network security policies. An information system has been said to have been "introduced" when it has been subjected to any kind of illegal action that leads to data corruption [15][16]. On the other hand, an intrusion is defined as any assault that can compromise the information's integrity as well as its secrecy [17]. The IDS can be categorized into two parts such as host-based and network-based as per the system deployment strategies. As a result, having IDS in place that can identify a variety of assaults is of the utmost significance [18]. So, an efficient IDS tool is required to safeguard significant data without any further damage. The techniques of detecting intrusions in the computer have captured the interest of a significant number of researchers [19][20]. The major contributions of this research are as follows: The IDS can be categorized into two parts such as host-based and network-based as per their deployment strategies

- This research mainly focuses on detecting DDoS attacks and enhancing the efficiency of IDS by utilizing the ResNet-50 and 3D CNN with the Black Widow Optimization (BWO) approach.
- ResNet-50 and 3D CNN are utilized for the classification process and it classifies the attacks into normal or anomaly.
- The proposed method's effectiveness is calculated

[1] School of CSE and IS, Presidency University, Bengaluru, India
ORCID ID : 0000-0002-8701-7686
[2] School of CSE and IS, Presidency University, Bengaluru, India
ORCID ID : 0000-0001-7747-4798
** Corresponding Author Email: teenakb1@gmail.com*

with various assessment metrics such as accuracy, precision, recall, and F1-score, by utilizing the benchmark datasets.

The rest of the paper is arranged as follows: Section 2 provides the literature survey. Section 3 presents the proposed methodology. Section 4 provides the results and discussion and Section 5 presents the conclusion of this research paper.

2. Literature Survey

Zohre Majidian et al. [21] introduced the vast variety of network infrastructure technologies and devices that may be compromised by a Denial of Service (DoS) attack. The suggested approach identified a way to use ECOC based hybrid classifiers to deal with the problem of the huge number of classes in the IDS. Features are extracted using principal component analysis (PCA) and then classified using a hybrid of Error-Correcting Output Codes (ECOC) as well as an Adaptive Neuro-Fuzzy Inference System (ANFIS). An ANFIS structure was optimized by the Particle Swarm Optimization (PSO) method to attain optimal performance in this classification model. The findings also reveal that the suggested strategy was far more accurate than the old one at detecting the various forms of DoS assaults. However, the suggested approach easily fell into local optima in high-dimensional space as well as a low convergence rate.

Abdullah Alzaqebah et al. [22] implemented a Grey Wolf Optimization algorithm (GWO), a tweaked bio-inspired algorithm designed to enhance the IDS's capacity to detect anomalies in network data. A smart initialization phase combined filter as well as wrapper techniques to guarantee that an informative characteristic was involved in prior iterations. Furthermore, the approach employed the Modified GWO to fine-tune the settings of the high-speed Extreme Learning Machine (ELM) classification algorithm. The suggested approach aimed to detect the generic attacks in network traffic which was the most significant attack. However, the balance between exploitation as well as exploration was being trapped in local optima.

Shubhra Dwivedi et al. [23] developed the combination of Ensemble Feature Selection with Grasshopper Optimization Algorithm known as EFSGOA. Initially, the EFS approach was used to grade characteristics in preparation for identifying the most important aspects. The EFS approach generated a reduced characteristics set that was then fed into GOA to help find key traits that may help identify the attack type. The GOA utilized the Support Vector Machine (SVM) as the fitness function to acquire important features as well as to optimize the penalty factor, kernel parameter as well as tube size parameters for enhanced classification performance. However, the suggested approach required a larger training time as well as low detection accuracy.

Smitha Rajagopal et al. [24] presented the meta-classification approach that utilized the decision jungle to perform the binary as well as multiclass classification. The suggested approach established robustness through an organization of optimal hyper-parameter sets associated with relevant feature subsets. The characteristic capability of the suggested approach was performed in its diplomacy to efficiently detect the 33 modern attack types. The suggested approach performed the statistical significance tests to compare the classifiers' performance involved in that approach. However, the suggested approach performance was not too efficient when compared to multiclass Logistic Regression (LR) as well as Decision Forest (DF).

Juan Fernando Canola Garcia and Gabriel Enrique Taborda Blandon [25] implemented the DL approach to design the IDS/Intrusion Prevention System (IPS) known as Dique, which could detect as well as prevent DoS attacks. The ID/IPS utilized the multi-layered Deep Feed Forward Neural Network (DFNN) approach and classified incoming packets to the web server into two classes such as benign as well as malicious. Dique had a Graphical User Interface (GUI) where “real life” can show textually as well as graphically the captured data as well as classify the packets, which permitted to move among IDS as well as IPS mode of the system operation. However, the suggested method required multiple lightweight intrusion detection systems for efficient classification.

Mahalakshmi G et al. [26] introduced the DL approach of CNN to solve the problem of intrusion identification in the network. The data had been normalized as well and categorical features were converted into numerical data from the data encoding approach for the data pre-processing approach. The UNSW-NB standard dataset was utilized for the CNN training process and the dataset involves binary classes of 0 and 1 for normal and attacks. However, the features of the dataset generated do not represent the network behavior in a diverse environment.

BC Preethi et al. [27] presented the Cycle-consistent Generative Adversarial Network (CCGAN) optimized by Water Strider Optimization (WSO) adopted IDS for protected Cloud Computing (CC) background (IDS-CC-CCGAN-WSOA). A CCGAN approach did not express the variation of optimization plans to measure optimal variables as well as guarantee suitable intruder detection in CC. The data was characterized as normal as well as anomalous data through the support of CCGAN. However, the results of the sensitivity as well as specificity were essential to improve the performance.

From the overall analysis, the existing methods have limitations such as enhanced training time of learning approaches, which resulted from the ANFIS approach through PSO. The balance between exploitation as well as exploration was being trapped in local optima. The features of the dataset generated do not represent the network behavior in a diverse environment. To overcome these limitations, this research proposes the ML-based classification algorithm with BWO to enhance the performance as well as accuracy results.

3. Proposed Methodology

The ML approach is utilized to identify the attacks in DDoS, which aids in the model's improvement to learn automatically. The proposed method involves four important blocks such as dataset collection, pre-processing, feature selection and classification. Fig. 1 depicts the block diagram for the proposed methodology.

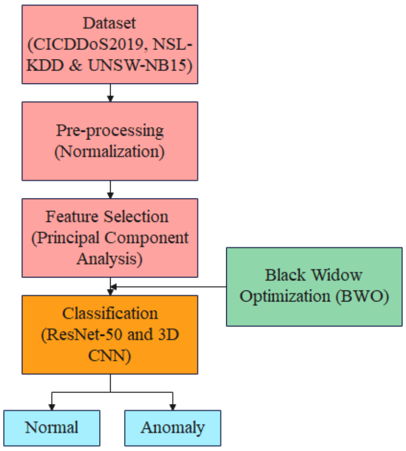
Dataset (CICDDoS2019, NSL-KDD & UNSW-NB15)
Pre-processing (Normalization)
Feature Selection (Principal Component Analysis)
Black Widow Optimization (BWO)
Classification (ResNet-50 and 3D CNN)
Normal
Anomaly

Fig 1. Block diagram of the proposed method

3.1. Dataset

The initial stage of the proposed method focuses on the dataset, which is utilized for the method pre-processing. An Intrusion Detection System (IDS) has used the three datasets to test ML methods in the classification of attacks of distributed Denial of Service (DDoS). The datasets utilized for the research of detecting the DDoS attacks in IDS are the CICDDoS2019, Network Socket Layer–Knowledge Discovery in Databases (NSL-KDD) and UNSW-NB15 datasets. The detailed explanation of these two datasets is described below.

3.1.1. CICDDoS2019

The CICDDoS2019 [28] subset is selected for this proposed work, which provides two variant detection schemes such as intrusion labels of four or two (normal and attack). The DDOS attacks are divided into 11 subtypes and every record is described by 86 features. This dataset is split into binary stages such as connections of training for 80% (539579) as well as testing 20% (134884) respectively.

3.1.2. NSL-KDD

NSL-KDD [29] is introduced to overcome the problem in the KDDCUP99 dataset. This data is built for the training (1,25,973) and testing (22,544) data samples. The advantage of this dataset is that it removes the redundant data from a train. This dataset contains 5 classes such as normal and four forms of attacks like DoS attack, probe, R2L, and U2R and also contains the 41 attributes.

3.1.3. UNSW-NB-15

The UNSW-NB15 [30] dataset is the raw network packet, which is acquired from IXIA Perfect-Storm tool for Cyber Security to produce real-time modern activities and behaviors of the attack. In training the dataset, 175,341 records are used and it decreases to 82,332 records in testing the dataset. This dataset contains nine different attack types Analysis, Worms, Backdoors Reconnaissance, Generic, Fuzzers, Shellcode, and DoS, Exploits.

The description of CICDDoS2019, NSL-KDD as well as UNSW-NB15 datasets are analyzed and these datasets are then provided to the pre-processing step and must be utilized to facilitate the categorization process.

3.2. Pre-processing

The pre-processing step utilized a collected intrusion dataset as the input. The pre-processing of data is an important part of cleaning text data which helps to examine the attacks in Distributed Denial of Service (DDoS). The purpose of pre-processing is to convert the rough data into a pattern, which is more effective to the ML method as well as improves model quality. The data will be normalized and the features of the category are converted into the numerical method by utilizing the method of encoding data.

3.2.1. Normalization

The normalization [31] is the pre-processing method, that is mainly utilized in the preparation of data part for the models of ML. The normalization aims to regulate the numerical data values in the dataset (i.e., 0 and 1) by utilizing the basic scale without the pervert variation in the original value ranges or else losing data. The normalization can be utilized by designing a new value that can control the basic distribution and the resource data ratios to protect the values in that the scale is adapted to all numeric columns utilized in the model. The attributes are normalized in the range based on the succeeding (1) as

$$Z_{max} = Z - \frac{\min(x)}{\max(z)} - \min(z) \qquad (1)$$

Where, min (z), max (z) – minimum, maximum values of an attribute Z; Z – the original feature value of Z and Z_{max} – normalized feature value of Z.

In this step, the preprocessing stage designs each jobless data and reduces the variance of the data, and then normalized data is forwarded to the feature selection process.

3.3. Feature Selection

The pre-processed output is provided as the input for the process of feature selection and it is also known as the selection of attribute. It is the process of selecting the most important features from the pre-processed data to enhance the model performance. The selection of features reduces the training time as well as eliminates the computational complexity. The feature selection technique is a dimensionality reduction process, that can be utilized to eliminate the non-applicable attributes in larger dimensional data named information of Intrusion Detection System (IDS). This method used the analysis of principal components for the process of feature selection.

3.3.1. Principal Component Analysis (PCA)

The Principal Component Analysis (PCA) [32] is largely utilized in an unsupervised approach for the selection of features. This PCA aims to minimize the dimensionality by protecting the important attributes data in the collection of datasets as well as selecting the significant feature subset in the dataset for clarification. This method utilized the PCA to reduce the space of the dataset of CICDDoS2019. The first element values are expressed in (2) as

$$Y_1 = Lb_1$$

(2)

Where, L – observation matrix with a zero mean and b – vector with the larger variance (y).

3.4. Black Widow Optimization

The BWOA [33] is a bio-inspired meta-heuristic optimization technique. By using BWOA, the search area is ensured to be investigated and used by modeling the numerous movement strategies utilized by spiders throughout the courtship and mating processes. Fig. 2 shows the workflow of the BWOA algorithm. The first generation of spiders tries to have babies by mating with itself. The female black widow eats the male in processing or after mating. The male then transports the sperm in its sperm membranes to the egg sacs. There is evidence of cannibalism among siblings when they are forced to share the accommodations of the maternal web in 11 days.

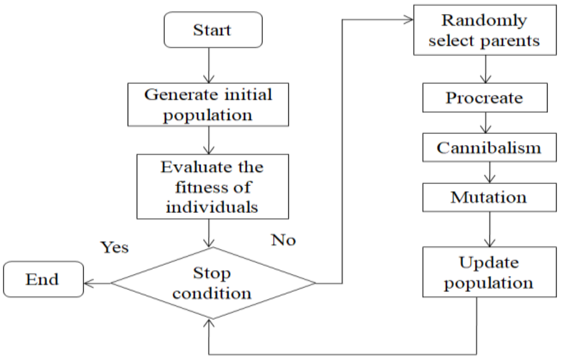

Fig 2. Workflow of BWO

3.4.1. Initial population

The initial black widow spider population in the black widow optimization is generated at random and consists of both male and female black widow spiders. Using an assessment of the fitness function f at a widow, determining the widow fitness for a given black widow spider population, which is expressed in (3) and (4) as.

$$X_{N,d} = \begin{bmatrix} x_{1,1} & x_{1,2} & x_{1,3} \dots \dots x_{1,d} \\ x_{2,1} & x_{2,2} & x_{2,3} \dots \dots . x_{2,d} \\ x_{N,1} & x_{N,2} & x_{N,3} \dots \dots . . x_{N,d} \end{bmatrix} \qquad (3)$$

$$lb \leq X_i \leq ub \qquad (4)$$

Where, lb is the population lower bound, $X_{N,d}$ is the black widow spider's population, N is the number of populations, and ub is the population upper bound, and d- decision variables numbers of the problem. To minimize or maximize a value of an objective function shown in Equation 2, the possible solution populations are employed in (5) as.

$$Objective\ function = (X_{N,d}) \qquad (5)$$

3.4.2. Procreate

As the couples are independent of one another, they may begin mating simultaneously to generate offspring. In nature, however, each pair of spiders mates alone in its web. A real-life spider couple may expect to generate about a thousand eggs, but only the hardiest of the offspring will make it. The algorithm's offspring are formed by plugging a long array containing random integers into an existing array named widow and multiplying by using µ. Here, $x1$ and $x2$ are the parents, while $y1$ and $y2$ are the offspring, which are expressed in (6) and (7) as.

$$y1 = \mu \times x1 + (1 - \mu) \times x2 \qquad (6)$$

$$y2 = \mu \times x2 + (1 - \mu) \times x1 \qquad (7)$$

Where $y1$ and $y2$ are the offspring of the spider colony, i and j are random integers from 1 to N, µ and is a random value between 0 and 1.

3.4.3. Cannibalism

There are three types of cannibalism in action here: The first is a kind of cannibalism that occurs during processing or after mating when a female black widow consumes her spouse. With this technique, gender is determined only by fitness metrics. The powerful spider lings will eat their siblings at processing or after mating. By assigning a cannibalism rating (CR) in this system, predicting how many people will make it. One common example of another kind of cannibalism is the young spiders eat their parents. A fitness value is utilized for separating the powerful spider lings from the weak ones.

3.4.4. Mutation

At this point, choose a random Mute pop number from the population of individual forms. Each of the optimal answers shown in Fig. 3 involves a random swap of two items of the array. Mutation frequency is used to estimate the mutated population size.

3.4.5. Convergence

When using an evolutionary algorithm, it is possible to choose between three distinct termination criteria: (1) a predefined number of iterations. (2) The best widow's fitness level has been stable across several iterations and (3) Obtaining the desired level of accuracy.

3.5. Classification

The eventual aim of the classification approach is to attain the greatest probable classification accuracy. In this research, the DL approach of the ResNet50 and 3D CNN algorithm is used for the classification process.

3.5.1. ResNet50

The ResNet50 is a combination of 48 convolutional layers with one average pool and one max pool layer. The ResNet50 can increase the training speed, maximize the network depth without the need for any parameters, and minimize the vanishing gradient problems effect. This ResNet50 contains five stages.

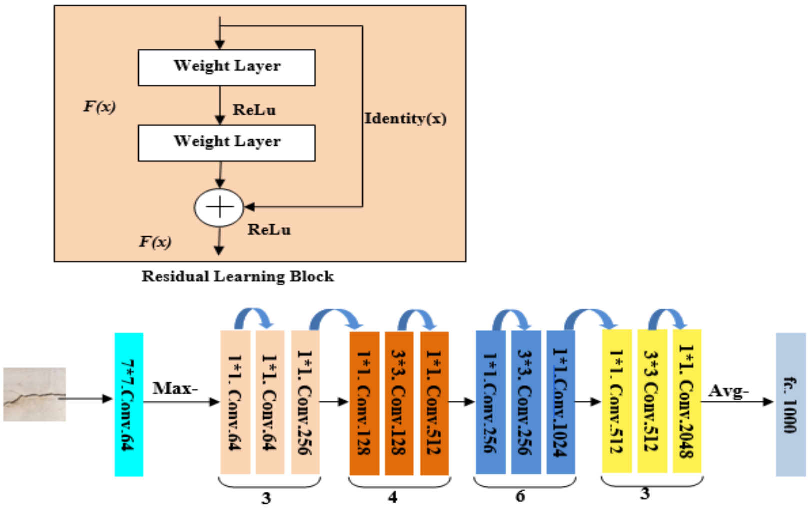

Fig 3. Resnet50 architecture

Fig. 3 depicts the general ResNet50 architecture. The input stage is the initial stage, which contains only one convolutional layer with batch normalization and the initial feature map is generated by using the activation function. The remaining states have identity blocks and convolutional blocks. These two blocks contain a convolutional layer with activation functions and additional batch normalization. To increase the residuals of the convolutional blocks, the input layer has an extra bridge to the output layer. The residual block on Resnet-50 is represented in (8) as,

$$y = F(x, W + x) \qquad (8)$$

Where, x and y represent the input and output layer respectively, F - residual map. The residual block on the Resnet-50 is proficient when the input data is indistinguishable from the output data.

3.5.2. 3D Convolutional Neural Network (3D CNN)

The 3D CNN [34][35] is a learning model, that is utilized for the IDS classification. The 3D CNN is made up of a variety of neurons with learning weights and biases. It is a feed-forward neural network in which, the artificial neurons can react to the surrounding units, as well as it will achieve the best results in the classification of image as well as recognition. This 3D CNN majorly includes the three structures such as,

- A convolutional layer for feature extraction.
- A pooling layer is utilized for the reduction of feature dimensionality.
- A fully connected layer (FC) is majorly used for the process of classification.

The data features are taken out along the convolutional layer after the dimensionality is reduced by the pooling layer features. The 3D CNN eliminates the risk of over-fitting, as well as improves the method's accuracy and fault tolerance. The features are acquired by the convolution and pooling layers are classified in the fully connected layer. The 3DCNN receives a greater number of inputs carry the weighted sum of inputs and issue them to the activation function to the given output. The 3D CNN will train on the 3D tensors acquired from reconstructed phase space representation. In dissimilarity to traditional pattern recognition systems, the CHH has generally attached the feature selection as well as classification levels in an end-to-end approach. Therefore, 3D CNN has been examined using the architecture that trains 3D tensor features are a new depiction of the input data. After, it gives weights to an outcome of filters to highlight or eliminate the revealing or jobless features correspondingly. The weights are implemented by the training process by reducing the loss function trying to connect the input data to a target class outcome. Fig. 4. Shows the general architecture of 3D CNN.

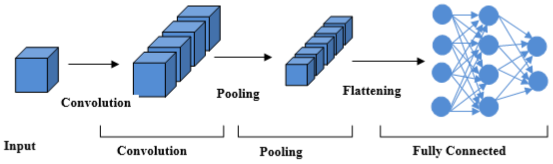

Fig 4. Architecture of 3D CNN

An input is a 3D tensor with a $256 \times 256 \times 256$ size. Though a smaller input size can minimize a resolution as well as subsequently cause data loss, the data in a larger size can create complex computations. The 3D CNN approach involves 5 convolutional layers by 64, 128, 256, 512 as well as 512 3D filters. A kernel size of convolutional layers $is\ 3 \times 3 \times 3$ with stride 1 and max pooling with the size of $2 \times 2 \times 2$ as well as stride size of 2 every convolutional layer and it also minimizes the dimensions of the features. Eventually, a destruction as well as two fully connected layers are placed later. The destruction layer exchanges the 3D tensor as a vector. An initial FC (FC1) involves 1000 neurons as well as final FC (FC2) is a classification layer. Every FC layer utilizes the Rectified Linear Unit (ReLU) activation function to develop nonlinearity into the model. The Rectified Linear Unit (ReLU) activation function is expressed in (9) as:

$$ReLU(x_i) = \begin{cases} x_i\ x_i \geq 0 \\ 0\ x_i < 0 \end{cases} \qquad (9)$$

Where, x_i - ith input to convolutional layer; the SoftMax activation function is performed in (10) as;

$$softmax(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \qquad (10)$$

Where, z_i and K – values as well as neuron numbers from the final layer. After the selection of efficient features, the classification approach has been trained to identify the intrusions from traffic areas. In this research, the DL-based classification algorithm has efficiently classified the IDS attacks into normal or anomaly

.

4. Experimental Results

Implementation as well as estimation of the proposed model are performed by MATLAB on an Intel core i5 processor, windows 10 OS, 6GB GPU, and 1TB SSD. The performance of the proposed method has been estimated utilizing DDoS attack illustrations of CICDDoS2019, NSLKDD and UNSW-NB15 datasets. Every data record in datasets involves 42 attributes. Utilizing PCA, the data dimensions of every data is minimized to 10 features. The appropriate number of features is acquired for experimenting by using various performance metrics. The number of PCs which leads to greater classification accuracy, was examined for various features. The various performance matrices have been performed to evaluate the proposed method's performance as well as effectiveness. The mathematical expressions of performance metrics such as accuracy, specificity, precision, recall, F1-score, Mean Absolute Error (MSE) as well as Mean Square Error (MSE), are provided in (11) to (17) as:

$$Accuracy = \frac{TP+TN}{FP+TP+FN+TN} \qquad (11)$$

$$Specificity = \frac{TN}{FP+TP} \qquad (12)$$

$$Precision = \frac{TP}{FP+TP} \qquad (13)$$

$$Recall = \frac{TP}{FN+TP} \qquad (14)$$

$$F1-score = \frac{2TP}{2TP+FP+FN} \qquad (15)$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i) \qquad (16)$$

$$MAE = \sqrt{\frac{1}{m}}\sum_{i=1}^{n}|y_i - \hat{y}_i| \qquad (17)$$

Where, TP, TN, FP and FN are True Positive, True Negative, False Positive and False Negative. n - number of points; y_i - predicted value obtained from the neural network; $\hat{y}_i$ - the real value and $\bar{y}$- mean of the real value.

4.1. Performance Analysis

This section represents the quantitative and qualitative analysis of the proposed BWO-based SVM approach about accuracy, precision, recall, F1-score, MSE, and MAE. Table 1 depicts the classification results using the CICDDoS2019 dataset. Table 2 denotes the classification results using the NSL-KDD dataset. Table 3 denotes the classification results utilizing the UNSW-NB15 dataset.

Table 1. Classification results using the CICDDoS2019 dataset

Method	Accuracy	Precision	Recall	F1-score	MSE	MAE
ResNet50	0.8623	0.8303	0.8368	0.8098	0.14	0.16
CNN	0.8878	0.8619	0.8419	0.8322	0.12	0.14
2D CNN	0.9175	0.8688	0.8682	0.8684	0.12	0.11
3D CNN	0.9287	0.8897	0.8794	0.9056	0.09	0.09
ResNet50-3D CNN	0.9321	0.9456	0.9623	0.9527	0.08	0.08

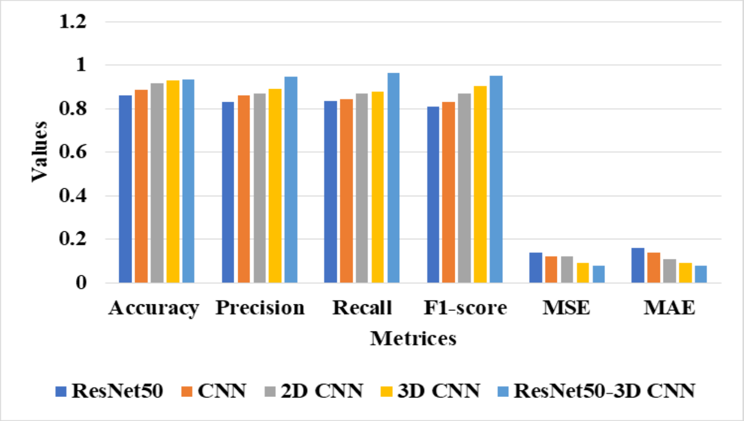

Fig 5. Graphical representation of performance analysis using CICDDoS2019 dataset

Table 1 and Fig. 5 represent the performance analysis of the proposed method using the CICDDoS2019 dataset. The performance of existing methods like ResNet50, CNN, 2D CNN as well as 3D CNN are measured and compared with the proposed ResNet50-3D CNN method. The proposed ResNet50-3D CNN method achieves the classification of 0.9321, precision of 0.9321, recall of 0.9623, F1-score of 0.9527, MSE of 0.08, and MAE of 0.08 respectively.

Table 2. Classification results using NSL-KDD dataset

Method	Accuracy	Precision	Recall	F1-score	MSE	MSE
ResNet50	0.9012	0.8587	0.8489	0.8137	0.22	0.23
CNN	0.9122	0.8711	0.8521	0.8509	0.17	0.21
2D CNN	0.9323	0.8738	0.8738	0.8948	0.9	1.10
3D CNN	0.9512	0.8917	0.8934	0.9146	0.07	0.08
ResNet50-3D CNN	0.9912	0.9934	0.9912	0.9912	0.06	0.05

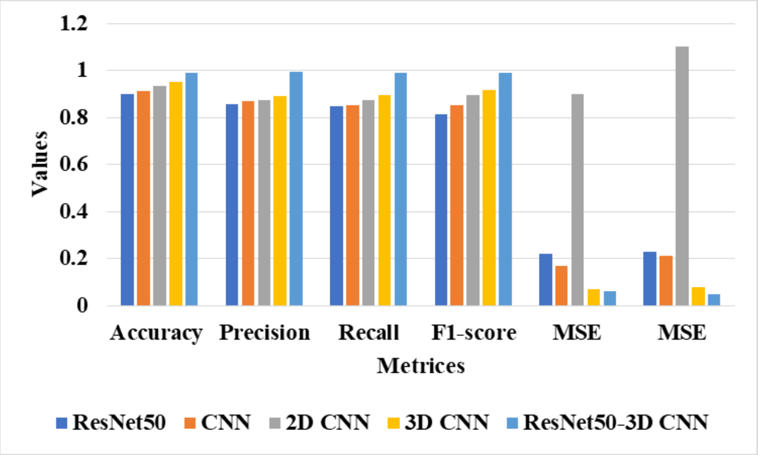

Fig 6. Graphical representation of performance analysis using the NSL-KDD dataset

Table 2 and Fig. 6 represent the performance analysis of the proposed method using the NSL-KDD dataset. The performance of existing methods like ResNet50, CNN, 2D CNN as well as 3D CNN are measured and compared with the proposed ResNet50-3D CNN method. The proposed ResNet50-3D CNN method achieves the classification of 0.9912, precision of 0.9934, recall of 0.9912, F1-score of 0.06 MSE of 0.06, and MAE of 0.05 respectively.

Table 3. Classification results using the UNSW-NB15 dataset

Method	Accuracy	Precision	Recall	F1-score	MSE	MAE
ResNet50	0.8756	0.8329	0.8198	0.8098	0.14	0.12
CNN	0.8911	0.8598	0.8321	0.8439	0.12	0.11
2D CNN	0.9146	0.8657	0.8648	0.8826	0.09	0.09
3D CNN	0.9412	0.8829	0.8745	0.9046	0.08	0.08
ResNet50-3D CNN	0.9634	0.9523	0.9622	0.9633	0.07	0.07

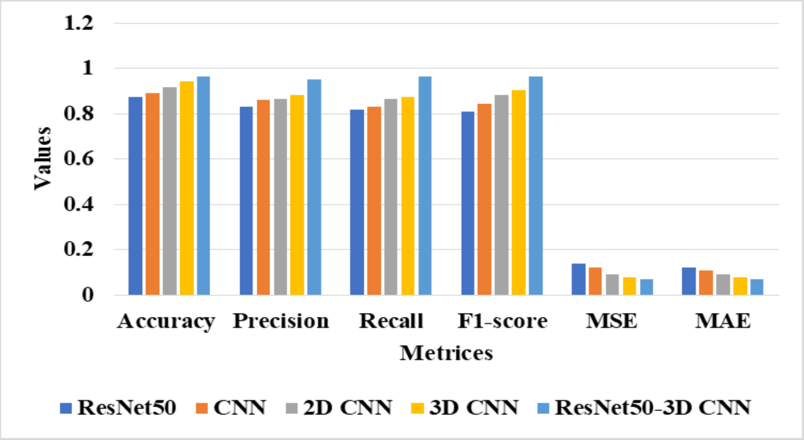

Fig 7. Graphical representation of performance analysis using the UNSW-NB15 dataset

Table 3 and Fig. 7 depict the performance analysis of the proposed method with regard to using the UNSW-NB15 dataset. The performance of existing methods like ResNet50, CNN, 2D CNN as well as 3D CNN are measured and compared with the proposed ResNet50-3D CNN method. The proposed ResNet50-3D CNN method achieves the classification of 0.9634, precision of 0.9523, recall of 0.9622, F1-score of 0.9633, MSE of 0.07 and MAE of 0.07 respectively.

Table 4. Overall performance Analysis of classification results

Method	Accuracy	Precision	Recall	F1-score	MSE	MAE
BWO-NB	0.8836	0.8539	0.8290	0.88347	0.18	0.18
BWO-LR	0.9045	0.8731	0.8419	0.8548	0.14	0.17
BWO-SVM	0.9367	0.8855	0.8858	0.8933	0.12	0.15
BWO-AdaBoost	0.9528	0.8936	0.8946	0.9157	0.09	0.13
BWO- ResNet50-3D CNN	0.9737	0.9218	0.9038	0.9239	0.08	0.09

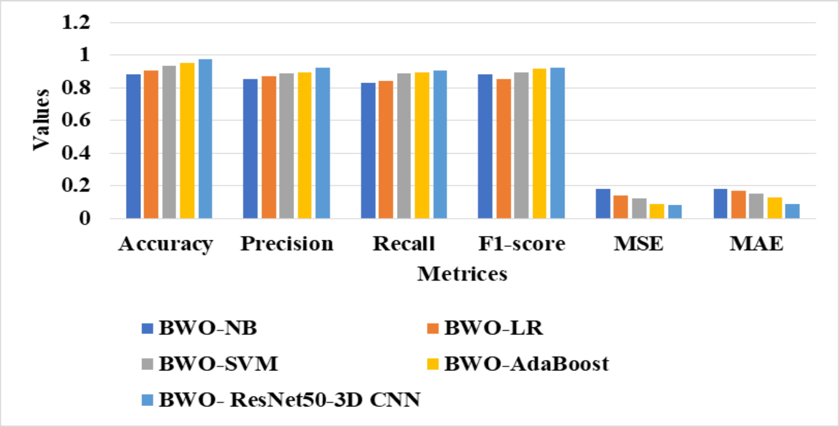

Fig 8. Graphical representation of overall performance analysis

Table 4 and Fig. 8 depicts the overall performance analysis of the proposed method. The performance of existing methods like ResNet50, CNN, 2D CNN as well as 3D CNN are measured and compared with the proposed ResNet50-3D CNN method. The proposed ResNet50-3D CNN method achieves the classification of 0.9737, precision of 0.9218, recall of 0.9038, F1-score of 0.9239, MSE of 0.08 and MAE of 0.09 respectively.

4.2. Comparative Analysis

Table 5 shows the comparison results of the proposed method. The proposed method's performance is evaluated by utilizing evaluation metrics like accuracy, precision, recall, F1-score, MAE, and MSE.

Table 5. Comparison of the proposed method with existing methods

Method	Dataset	Accuracy	Precision	Recall	F1-score	MAE	MSE
DFNN [20]	CICDDoS2019	0.91	0.93	0.9579	0.9464	N/A	N/A
CNN [21]	UNSW -NB15	0.95	N/A	N/A	N/A	0.09	0.09
IDS-CC-HDLNN-WSOA [22]	NSL-KDD	0.98	0.98	0.98	0.98	N/A	N/A
Proposed Resnet50-3D CNN- BWO	CICDDoS2019	0.9321	0.9456	0.9623	0.9527	0.08	0.08
	UNSW -NB15	0.9634	0.9523	0.9622	0.9633	0.07	0.07
	NSL-KDD	0.9912	0.9934	0.9912	0.9912	0.06	0.05

4.3. Discussion

This section provides the information on limitations of the existing methods and the advantages of the proposed method. The limitations are the DFNN [20] requires multiple lightweight intrusion detection systems for efficient classification. In CNN [21], the generated features do not represent the network behavior in various environments. In IDS-CC-HDLNN-WSOA [22], the balancing between exploitation and exploration was trapped in local optima in high dimensional data. The proposed ML-based classification with BWO outperforms these existing model limitations. The BWO is good for both local exploitation and global exploration. The BWO obtains a better balance between exploration as well as exploitation, improving convergence speed. The proposed method depicted the significant outcomes with almost 99% accuracy. The proposed method is quantified for the specific type of attacks as well and it achieves more than 90% accuracy by utilizing the 17 features and it depicted faster convergence compared to the standard BWO.

5. Conclusion

IDS have emerged as a crucial component of modern information security measures because the number of attacks has skyrocketed with the expansion of the internet. The suggested approach that recognizes intrusion detection is separated into normal and abnormal labels. According to the findings, the proposed method is more efficient and shows better performance in accuracy, precision, recall, and F1-score compared to existing methods. In subsequent research, the primary attention may shift to the formulation and validation of alternative alterations to improve meta-heuristic methods to feature selection issues in general and the intrusion detection system in particular. In further development, the proposed method will extend to evaluate the system by using original network data. Real-time data may be created and gathered with the assistance of many attack simulation tools, which can cover a variety of newly emerging assaults. In the future, approaches including deep learning may be used to manage real-time attacks.

Author contributions

K.B. Teena: Conceptualization, Methodology, Software, Field study, Data curation, Writing-Original draft preparation, Software, Validation., Field study **Swati Sharma:** Visualization, Investigation, Writing-Reviewing and Editing.

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] G. Logeswari, S. Bose, and T. Anitha, "An intrusion detection system for sdn using machine learning," *Intell. Autom. Soft Comput.*, vol. 35, no. 1, pp. 867–880, Jan. 2023.

[2] Z. Wu, H. Zhang, P. Wang, and Z. Sun, "RTIDS: A robust transformer-based approach for intrusion detection system," *IEEE Access*, vol. 10, pp. 64375–64387, Jun. 2022.

[3] R. Mangayarkarasi, V. Chandrasekaran, V. Ravi, and N. Kumar, "An intrusion detection system using optimized deep neural network architecture," *Trans. Emerging Telecommun. Technol.*, vol. 32, no. 4, p. e4221, Feb. 2021.

[4] G. Apruzzese, L. Pajola, and M. Conti, "The cross-evaluation of machine learning-based network intrusion detection systems," *IEEE Trans. Netw. Serv. Manage.*, vol. 19, no. 4, pp. 5152–5169, Dec. 2022.

[5] M. Vishwakarma and N. Kesswani, "DIDS: A Deep Neural Network based real-time Intrusion detection system for IoT," *Decision Analytics Journal*, vol. 5, p. 100142, Dec. 2022.

[6] T. Saba, A. Rehman, T. Sadad, H. Kolivand, and S. A. Bahaj, "Anomaly-based intrusion detection system for IoT networks through deep learning model," *Comput. Electr. Eng.*, vol. 99, p. 107810, Apr. 2022.

[7] M. Soltani, B. Ousat, M. J. Siavoshani, and A. H. Jahangir, "An adaptable deep learning-based Intrusion Detection System to zero-day attacks," *J. Inf. Secur. Appl.*, vol. 76, p. 103516, Aug. 2023.

[8] S. A. Hussein, A. A. Mahmood, and E. O. Oraby, "Network intrusion detection system using ensemble learning approaches," *Technology*, vol. 18, pp. 962–974, Oct. 2021.

[9] Y. Otoum, D. Liu, and A. Nayak, "DL-IDS: a deep learning–based intrusion detection framework for securing IoT," *Trans. Emerging Telecommun. Technol.*, vol. 33, no. 3, p. e3803, Mar. 2022.

[10] M. A. Khan and Y. Kim, "Deep learning-based hybrid intelligent intrusion detection system," *Comput. Mater. Contin*, vol. 68, pp. 671–687, Sep. 2023.

[11] G. N. Tikhe, and P. S. Patheja, "A wrapper feature selection based hybrid deep learning model for ddos detection in a network with nfv behaviors," *Wireless Pers. Commun.*, vol. 133, pp. 481–506, Dec. 2023.

[12] T. Gaber, J. B. Awotunde, M. Torky, S. A. Ajagbe, M. Hammoudeh, and W. Li, "Metaverse-IDS: Deep learning-based intrusion detection system for Metaverse-IoT networks," *Internet Things*, vol. 24, p. 100977, Dec. 2023.

[13] V. Ravi, R. Chaganti, and M. Alazab, "Recurrent deep learning-based feature fusion ensemble meta-classifier approach for intelligent network intrusion detection system," *Computers and Electrical Engineering*, vol. 102, p. 108156, Sep. 2022.

[14] V. Hnamte, H. Nhung-Nguyen, J. Hussain, and Y. Hwa-Kim, "A Novel Two-Stage Deep Learning Model for Network Intrusion Detection: LSTM-AE," *IEEE Access*, vol. 11, pp. 37131–37148, Apr. 2023.

[15] R. Chaganti, W. Suliman, V. Ravi, and A. Dua, "Deep learning approach for SDN-enabled intrusion detection system in IoT networks," *Information*, vol. 14, no. 1, p. 41, Jan. 2023.

[16] A. Awajan, "A novel deep learning-based intrusion detection system for IOT networks," *Computers*, vol. 12, no. 2, p. 34, Feb. 2023.

[17] H. Wang and W. Li, "DDosTC: A transformer-based network attack detection hybrid mechanism in SDN," *Sensors*, vol. 21, no. 15, p. 5047, Jul. 2021

[18] M. Sarhan, S. Layeghy, N. Moustafa, M. Gallagher, and M. Portmann, "Feature extraction for machine learning-based intrusion detection in IoT networks," *Digital Communications and Networks*, Sep. 2022.

[19] C. Zhang, X. Costa-Perez, and P. Patras, "Adversarial attacks against deep learning-based network intrusion detection systems and defense mechanisms," *IEEE/ACM Trans. Networking*, vol. 30, no. 3, pp. 1294–1311, Jan. 2022.

[20] A. A. Megantara and T. Ahmad, "A hybrid machine learning method for increasing the performance of network intrusion detection systems," *J. Big Data*, vol. 8, no. 1, pp. 1–19, Nov. 2021.

[21] M. Zohre, S. TaghipourEivazi, B. Arasteh, and S. Babai, "An intrusion detection method to detect denial of service attacks using error-correcting output codes and adaptive neuro-fuzzy inference," *Computers and Electrical Engineering*, vol. 106, p. 108600, Mar. 2023.

[22] A. Abdullah, I. Aljarah, O. Al-Kadi, and R. Damaševičius, “A modified grey wolf optimization algorithm for an intrusion detection system,” *Mathematics*, vol. 10, no. 6, p. 999, Mar. 2022.

[23] D. Shubhra, M. Vardhan, and S. Tripathi, “Building an efficient intrusion detection system using grasshopper optimization algorithm for anomaly detection,” *Cluster Comput.*, vol. 24, pp. 1881–1900, Jan. 2021.

[24] S. Rajagopal, P. P. Kundapur, and K. S. Hareesha, “Towards effective network intrusion detection: from concept to creation on Azure cloud,” *IEEE Access*, vol. 9, pp. 19723–19742, Jan. 2021.

[25] J. F. C. Garcia and G. E. T. Blandon, “A deep learning-based intrusion detection and preventation system for detecting and preventing denial-of-service attacks,” *IEEE Access*, vol. 10, pp. 83043–83060, Aug. 2022.

[26] A. K. Samha, N. Malik, D. Sharma, and P. Dutta, “Intrusion detection system using hybrid convolutional neural network,” *Mobile Networks Appl.*, pp. 1–13, Aug. 2023.

[27] B. C. Preethi, G. Sugitha, and G. Kavitha, “Cycle-consistent generative adversarial network optimized with water strider optimization algorithm fostered intrusion detection framework for securing cloud computing environment,” *Concurrency Comput. Pract. Exper.*, vol. 35, no. 5, p. e7552, Feb. 2023.

[28] CICDDoS2019 dataset link: https://www.kaggle.com/datasets/dhoogla/cicddos2019.

[29] NSL-KDD dataset link: https://www.kaggle.com/code/avk256/nsl-kdd-anomaly-detection.

[30] UNSW-NB15 dataset link: https://www.kaggle.com/datasets/mrwellsdavid/unsw-nb15.

[31] V. Hnamte and J. Hussain, “DCNNBiLSTM: An efficient hybrid deep learning-based intrusion detection system,” *Telematics and Informatics Reports*, vol. 10, p.100053, Jun. 2023.

[32] S. A. Dheyab, S. M. Abdulameer, and S. Mostafa, “Efficient machine learning model for ddos detection system based on dimensionality reduction,” *Acta Informatica Pragensia*, vol. 11, no. 3, pp. 348–360, Dec. 2022.

[33] P. Dahiya and Saha A. K., “Frequency regulation of interconnected power system using black widow optimization,” *IEEE Access*, vol. 10, pp. 25219–25236, Feb. 2022.

[34] M. R. Falahzadeh, E. Z. Farsa, A. Harimi, A. Ahmadi, and A. Abraham, “3D Convolutional Neural network for speech emotion recognition with its realization on intel CPU and NVIDIA GPU,” *IEEE Access*, vol. 10, pp. 112460–112471, Oct. 2022.

[35] W. Xiang, X. Wu, C. Li, W. Zhang, and F. Li, “Driving fatigue detection based on the combination of multi-branch 3d-cnn and attention mechanism,” *Applied Sciences*, vol. 12, no. 9, p. 4689, May 2022.