Journal of Theoretical and Applied Information Technology
31st July 2023. Vol.101. No 14
© 2023 Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A NOVEL HYBRID INTRUSION DETECTION MODEL FOR INTERNET OF THINGS USING MACHINE LEARNING

L. SARALADEVE[1], A. CHANDRASEKAR[2]

[1]Research Scholar, Department of Computer Science, Bharathiar University, Coimbatore, Tamilnadu, India.

[2]Research Supervisor, Department of Computer Science, Bharathiar University, Coimbatore, Tamilnadu, India.

E-mail: [1]saraladeve@gmail.com

ABSTRACT

The identification of attacks in the infrastructure of the Internet of Things (IoT) is becoming a progressive problem in the field of IoTs. The proliferation of IoT infrastructures across all domains has coincided with an increase in the number of attacks and threats. To address this issue, a new hybrid intrusions detection system (IDS) model is developed in this paper for attacks detection and classification. The proposed model is a combination of a metaheuristic algorithm and a machine learning technique. The metaheuristic algorithm called Binary Enhanced-Whale Optimization Algorithm (BEWOA) is used for the selection of features and the machine learning algorithm called Random Nearest Neighbor (r-NN) is utilized for the classification. This BEWOA-rNN model is evaluated using the NSL-KDD and UNSW-NB-15 datasets. The workflow of the model includes the preprocessing, dataset splitting, feature selection and classification. Using the normalization technique, the preprocessing process is performed for data standardization and data cleaning. After preprocessing, the features from the datasets are selected using the BEWOA algorithm for improving the classification performance of the model. Based on the selected features the classification is performed and the performance is measured in terms of detection rate, accuracy, specificity, f-measure, and precision. The performance evaluation is measured individually for both the datasets using the BEWOA-rNN model. The model obtained 99.22% accuracy, 98.82% detection rate, 99.63% specificity, 99.64% precision, and 99.23% f-measure using the NSL-KDD dataset. The model obtained 99.06% accuracy, 98.90% detection rate, 99.22% specificity, 99.24% precision, and 99.07% f-measure using the UNSW-NB-15 dataset.

Keywords: IoT, Attack Detection, IDS, BEWOA, r-NN, Machine Learning.

1. INTRODUCTION

Millions of people now rely on the internet for a variety of reasons, and internet has evolved into a fundamental requirement in their lives. It is reported that more than sixty percent of the people living in the world make use of the internet. This indicates that more than half of the population makes use of the internet, which may be attributed to the widespread availability of the resource as well as the numerous advantages it offers to individuals. The IoT is a rapidly developing technology that is expanding as a result of the advantages offered by the internet. The IoT makes it possible for various devices and things to connect with and communicate with one another using the internet [1].

This technology was developed with the intention of automating formerly manual tasks and connecting the systems that people use on a daily basis using the internet. Each IoT system or thing has sensors affixed to it so that information can be gathered from the surrounding environment. Information is processed via local processing in order to eliminate data that is not necessary, and then the information is stored in local storages. The information was transferred from the local to the cloud, which is where each object submits the information that they have acquired. In the end, the relevant actions are carried out by making use of the information that was obtained. The action does not necessarily have to be carried out by making use of this knowledge in every instance. But users also have the ability to remotely manage and operate the equipment and objects, as well as utilize the data to keep records for potential use in future [2].

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The IoT network is quickly becoming an essential component of applications, such as smart homes, smart cities, smart grids, etc. People are given the ability to make intelligent decisions by way of machine-to-machine communication made possible by the developing network of smart devices. The IoT is expected to have almost 30 billion internet-connected devices by the year 2030, as stated by Statista (2022). The exponential growth of IoT devices is made possible by the internet, which enables their proliferation every day. The widespread implementation of IoT raises a number of important concerns, chief among them being security and privacy. Because of this, the IoT has not yet achieved widespread adoption [3].

The IoT is organized into three primary layers: the application layer, the network/transport layer, and the perception/physical layer. The first layer is the hardware layer, which is the perception layer. It includes and 6LowPAN, RFID, Bluetooth, which are examples of communication standards, as well as actuators and sensors that transmit and receives data. Next, the network layer was is responsible for ensuring that data is routed or transmitted in an efficient manner. Communication protocols such as Wi-Fi, GSM, 4G, 5G, IPv6, and others are utilized by it. Finally, the application layer, which is also known as the software layer, is the top layer that delivers user interfaces to end users and provides systems with the business logic that is required for the systems to function. As shown in figure 1 [4] every layer has the potential to represent numerous different types of vulnerabilities.

Several advantages of the IoT have become more readily available due to the proliferation of IoT applications. In contrast, it include problems like ineffective managements, insufficient energy efficiency, inadequate management of identity, inadequate privacy, and securities. The protection of personal information of users and data is the most pressing challenge facing the development of IoT. IoT network has nevertheless been exposed to network attacks, even with all the availability of typical security measures such as authentications, encryption, access controls, or data secrecy. As a result, a second line of defence is required to protect these networks from further intrusion. In circumstances like these, the significance of IDS for the IoT becomes relevant. The use of IDS is a common practice among the systems that make up the IoT.

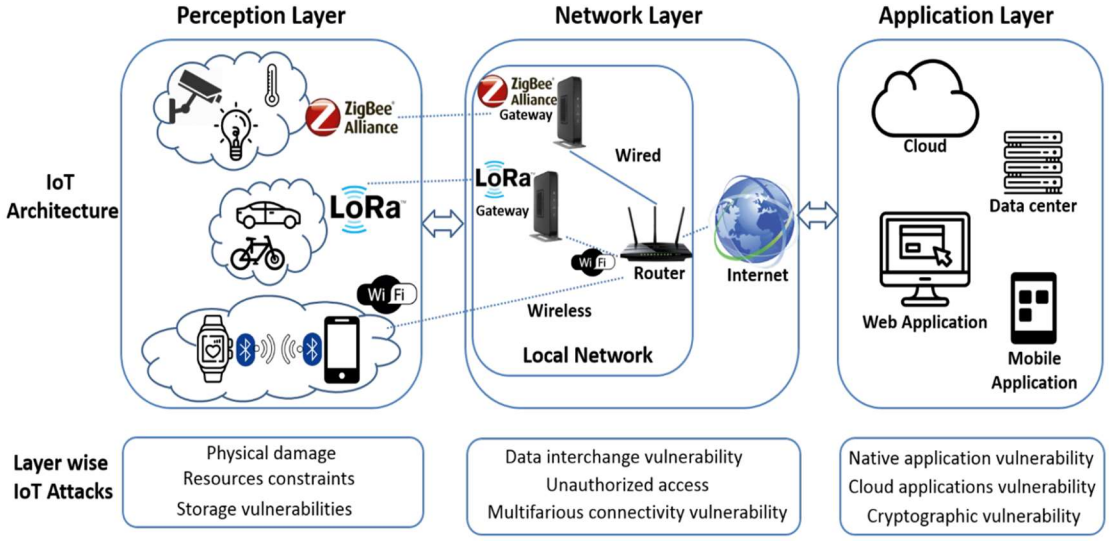

Figure 1: Architecture of IoT and Attacks in Each Layer

IDS is a potential security measure for the IoT. It does this by controlling and monitoring the activities of the network in order to observe how users behave. If there are environment changes, it will take preventative actions to unauthorized actions straight away. These defences are built into the various layers of the IoT to protect users' data from malicious users.

The IDS is a hardware and software combination that monitors computer networks or devices in order to determine potentially harmful activity and deliver instant notifications. IDSs are typically divided into different categories based on the deployment method and the detecting technique used. IDS deployments can be broken down into host and network IDS (NIDS). Installing host IDS

models requires a host machine, which could be a device or a Thing depending on the needs. They monitor the behaviours that are connected to system application files and operating systems, and they analyze the data they collect. The use of HIDSs is recommended for the purpose of insider intrusion prevention and deterrent. NIDS models are able to catch and examine the flow of packets in the networks. Specifically, they are inspecting packets that have been sniffed. The NIDS approach is effective against threats from external incursion [5]. The NIDS technique is narrowly targeted due to the fact that this research work is predicated on the security of IoT networks.

An effective detection system will identify the compromised condition and limit the loss by promptly recognizing the attacks. This is accomplished by identifying the threats in a timely manner. Misuse, Specifications, Anomaly, and Hybrid detection are the several categories that fall under the categories of IDS detection approaches.

- Signature or Misuse detection (knowledge-based) is the set of predetermined criteria that were given as input and coordinated with events.
- An event-based detection technique is also known by its more common name, anomaly detection. It outlines the typical activities that take place on the network, and any action that deviates from this pattern is considered an intrusion.

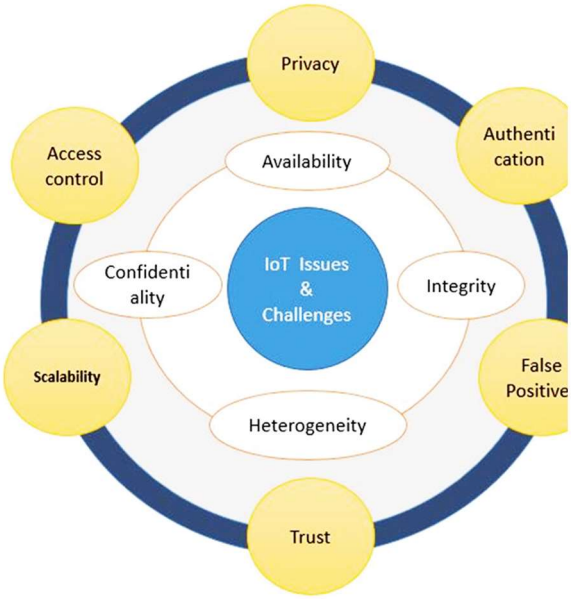

Figure 2: IoT Challenges & Issues

- The logic used for anomaly detection can also be used for specification detection. It describes an anomaly as a departure from the typical pattern of behaviour. In this scenario, the user physically defines what constitutes regular network activity, and the system will provide an alert if it detects any harmful behaviour within those activities. It takes significantly longer to complete than the anomalous technique.
- Hybrid detection is a combination of all three of the methods that were covered previously, most notably signature detection and detection based on anomalies. The accuracy of a hybrid detector can be improved by lowering the number of false positive events.

Developing an IDS is not a simple process since it includes a number of processes that need to be followed in order to deliver exact and valid outputs in the realm of IoT and associated applications. The primary purpose of this research is to identify attacks that can occur within IoT networks.

The following is a list of the contributions that this research work has made:

- In order to choose the most useful features and enhance the accuracy of the model, a metaheuristic algorithm known as BEWOA is utilized.
- For the purpose of classifying attacks within the IoT, a new method referred to as r-NN that was developed using the k-Nearest Neighbors (k-NN) model and combined with a metaheuristic algorithm has been proposed.
- Both the NSL-KDD and UNSW-NB15 datasets are utilized in the evaluation of the BEWOA-rNN model. The detection rate, accuracy, specificity, precision, and f-measure are the metrics that are utilized in order to evaluate the effectiveness of the research model.
- The performances of this research model are validated by comparing it with a number of different IoT-IDS models that have been examined from the related studies.

The issues over privacy and data protection have become increasingly important as the count of Internet-connected devices continues to grow. These issues are the primary impediments to the widespread adoption of the Internet of Things. Security in the internet of things has emerged as a primary concern. Although it is impossible to totally prevent attacks on any system in perpetuity, it is essential to detect attacks in real-time in order to keep systems protected in a convincing manner. Because of the numerous

security flaws in the IoT system, potential security attacks could be triggered by any application. In this way, there is a fundamental necessity for IDS to be implemented in IoT-based systems in order to protect against security threats that are dependent on weaknesses.

The following objectives provide a clear overview of the research goals and the specific tasks and evaluations conducted throughout the study.

- Propose a novel hybrid IDS model for detecting and classifying attacks in the IoT environment.
- Combine a metaheuristic algorithm (BEWOA) and a machine learning technique (r-NN) in the hybrid model.
- Evaluate the proposed model using two datasets, NSL-KDD and UNSW-NB-15.
- Perform preprocessing on the datasets, including data standardization and cleaning using normalization techniques.
- Utilize the BEWOA algorithm for feature selection to enhance the classification performance of the model.
- Measure the performance of the model using various metrics such as accuracy, detection rate, specificity, f-measure, and precision.
- Compare the performance of the proposed model with other existing models for validation purposes.
- Assess the model's performance individually for each dataset (NSL-KDD and UNSW-NB-15).
- Determine the effectiveness of the BEWOA-rNN model in classifying attacks in comparison to other models.

The remainder of this paper consists of the following sections organized as follows: The analysis of the relevant literature is provided in Part 2. The implementation of the research model is discussed in part 3. This discussion includes the presentation of BEWOA-based feature selection and rNN-based classification. In next part, the performances of the BEWOA-rNN model are analyzed and compared with a number of IDS models taken from the literature review. Finally, the conclusion is presented along with some suggestions for further research.

2. LITERATURE REVIEW

An increasing issue in the field of IoTs is the detection of attacks and anomalies in the infrastructures of IoT. In [6], a smart, safe, and dependable IoT-based model was established. This model was able to identify its own vulnerabilities, have a strong firewall against all cyberattacks, and automatically recover itself if it was compromised. Support Vector Machines, Random Forests, Logistic Regressions, Decision Trees, and Artificial Neural Network were the methods that were used to perform the prediction of anomalies and attacks in the IoT networks. Following the analysis of the data, it was determined that random forest model was superior than the other techniques. Despite the fact that the model was accurate 99.4% of the time, this work was carried out using data from a simulated environment. While dealing with data collected in real time, the model may experience a variety of difficulties.

An IDS model that makes use of a machine learning approach was proposed in [7] as a potential solution to the problems of insecurity, abnormality, and service failures that are prevalent in IoT contexts. An Energy Aware Smart Home (EASH) system was established as a result of the examination of the problems, and within this system, the problems with communication failures and the different types of network attacks were investigated. This model was derived from an IoT-based cyber-physical system in order to facilitate efficient communication through the application of efficient machine learning techniques. For the purpose of predicting assaults and errors, models like Naive Bayes, J48, Multinominal Logistics Regression, and Multilayer Perceptron were deployed. A dimensionality reduction strategy or a feature selection method could have been incorporated into this model in order to increase its accuracy, which was already at 85%.

A hybrid IDS model that is based on artificial bee colony and artificial fish swarm algorithms was developed in [8] to classify the normal and abnormal activities of network systems. The training dataset was partitioned using the fuzzy c-means clustering method, and characteristics that were deemed unnecessary were eliminated using the correlation-based feature selection method. In addition, the CART method was utilized to build If-Then rules in accordance with the selected record characteristics in order to differentiate between normal and anomalous data. This model was trained using the rules that were developed. The NSL KDD and UNSW NB-15 data sets were utilized in the process of implementing this model for detecting attacks. This model could have been improved in both the computational difficulty and the amount of time required by this model.

Intrusion detection has proven to be an effective method for securing networks because it can identify previously unknown attacks based on network traffic. A model of traffic anomaly

detection known as BAT was proposed in [9] as a potential solution to the issues of feature engineering and low accuracy that are prevalent in intrusions detection. This BAT model was a merger of bidirectional long-term memory and attention process. The attention mechanism was employed in order to observe the network flow vectors, which was made up of packet vectors produced by the bidirectional long short-term memory. As a result, the primary features necessary for the classification of network traffics were obtained. Additionally, several convolution layers were implemented so that the local characteristics of the traffic data could be extracted. During the processing of the data samples, multiple convolutional layers were utilized. The softmax classifier was utilized for the purpose of classifying the network traffic. This model was only accurate in identifying the time series data 84.25% of the time, which could have been improved.

In [10], an efficient detection system that makes use of the machine learning approach in order to detect Botnet attacks in IoT contexts was developed. As a detection model, a machine learning technique known as the classification and regression tree (CART) algorithm was used. The CART algorithm is one of the most well-known machine learning approaches that can be utilized to achieve high detection rates with minimal processing load for a system that protects against cyberattacks. The characteristics were gleaned from the incoming traffic patterns by the system. Following the collection of the necessary features, those features were utilized in the detection step of the process. During the detection phase, the CART algorithm was utilized to create the detection models and to classify the incoming patterns and features as either attacks or normal data. This was accomplished by comparing the incoming data to known good patterns and features.

In order to manage higher-dimensional and imbalanced network traffics, a framework for intrusions detection that was on the basis of features selection and ensembled learning models was developed in [11]. Dimensionality reduction was accomplished by implementing the correlations-based features selection with Bat algorithm. This selects the best possible subset of features based on the correlation that exists between those features. Combining the C4.5 model, the random forest model, and the forest by penalizing attributes model served as the basis for the ensemble method of attribute categorization. The probability distribution of the basis learner was combined for attack identification using, as a final step, a voting mechanism based on the probability's combination rules average. Using the subset of characteristics used for the CIC-IDS2017 data set, this model achieved better detection rate and accuracy possible.

In order to construct a data-driven intelligent IDS model, artificial intelligence, and in particular techniques from the domain of machine learning, can be utilized. As a result, a security model that is based on machine learning and is referred to as the Intrusion Detection Tree (IntruDTree) was proposed in [12]. This model began by evaluating the significance of the various security features and then proceeded to construct the tree-based standard IDS model according to the features that were deemed to be the most significant. This procedure was carried out in order to render this model reliable in relation to detection accuracy for test samples that had not previously been seen and efficient in terms of lowering the amount of computational cost incurred by processing the bare minimum of features. This model identified the abnormalities and normal class attacks according to the patterns in which they occurred in the security dataset, and as a result, it provided a significant outcome for test cases that had not previously been seen.

IDS systems that employ methods of machine learning are of the utmost significance for IoT networks. A two-stage hybrid method that used techniques from machine learning was developed in [13] for the purpose of detecting intrusions in the IoT network. In this part of the process, the genetic algorithm was utilized to choose relevant features in order to enhance the precision of this IDS framework. For the purpose of classification, machine learning models such as support vector machines, decision trees, and ensemble learning methods were applied. When compared to the results obtained by other classifiers, the ensemble classifier produced better results. The reason for this was because the ensemble used a variety of different learning strategies in order to achieve better results than any single method could have.

In [14], a lightweight IDS for 6LoWPAN networks based on the Routing Protocol for Low Power and Lossy Networks (RPL) was developed to identify wormhole attacks in the IoTs. This work presented three IDS models that are based on machine learning, including: K-means clustering unsupervised learning-based intrusion detection system, supervised decision tree-based intrusion detection system, and two-stage hybrid IDS that combined decision trees-based and K-means clustering approach for detecting the attacks in Internet of Things. The K-means methodology was successful in achieving a detection rate of 70-93% across a wide range of random IoT network sizes. For the similar sizes of network, the hybrid technique achieved the detection score of 71-75%, while the decision tree-based IDS achieved the

detection score of 71-80%. Even the hybrid IDS achieved a poor detection score than other techniques, it was more accurate than both of them. The hybrid technique produces a substantially lower amount of false positives compared to the other IDS models, which both produced a significantly high count of false positives.

The data-level technique and the feature-level approach were integrated in [15] construction of a hybrid approach, which resulted in the construction of a classifier ensemble-based IDS model. To enhance the detection rate of the IDS system, this model used machine learning. The ensemble technique made use of various machine learning classifiers, including J48, Logistic Regression, and Naive Bayes. In order to place more of an emphasis on the attack of uncommon categories, the ensemble technique additionally made use of the resampling technique. The utilization of preprocessing is beneficial to the overall performance of the classifier as it improves its ability to accurately recognize attacks of uncommon category types. However, there is a cost associated with it because the learning phase of the model takes significantly more time. Improving performance with relatively little overhead can be accomplished by dynamically adjusting the size of the dataset based on an analysis of the factors.

Overall, these previous works exhibit limitations in terms of real-world applicability, scalability, accuracy, computational complexity, and handling various types of attacks. The BEWOA-rNN model may have advantages in addressing these limitations, but further analysis and evaluation are required to validate its effectiveness in practical IoT environments.

3. METHODOLOGY

This research implemented a hybrid method based on the metaheuristic and machine learning algorithm to develop an IoT-IDS model. This proposed model enhances the accuracy of identifying intrusions (attacks). The workflow of the developed IDS model was represented in figure 3. NSL-KDD and UNSW NB-15, were applied as input to the model. As shown in the figure, the initial stage of the model performs the dataset preprocessing. The preprocessing stage performs the normalization procedure to standardize and remove the missing attributes in the datasets. After preprocessing, the datasets were labelled and split into training and testing sets. The feature selection was performed using the BEWOA algorithm using the training set, and the optimal features were selected. Based on the selected features, the classification was carried out using the r-NN algorithm, and the performance was measured using the detection rate, accuracy, f1-scores, specificity, and precision.

3.1. Description of Datasets

For the evaluation of this research, NSL-KDD and UNSW-NB-15 datasets are used as input to the proposed BEWOA-rNN model. Because the IoT platform encompasses traditional internet, mobile networks, non-IP networks, cloud computing, sensor networks and fog computing, a typical network dataset is not suitable for usage with IoT networks. These two datasets were the ones that were utilized the most for analyzing the IDS models of the IoT contexts. The objective was to identify the attacks according to the selected features with the r-NN classifier as either attack or normal type.

Journal of Theoretical and Applied Information Technology
31st July 2023. Vol.101. No 14
© 2023 Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

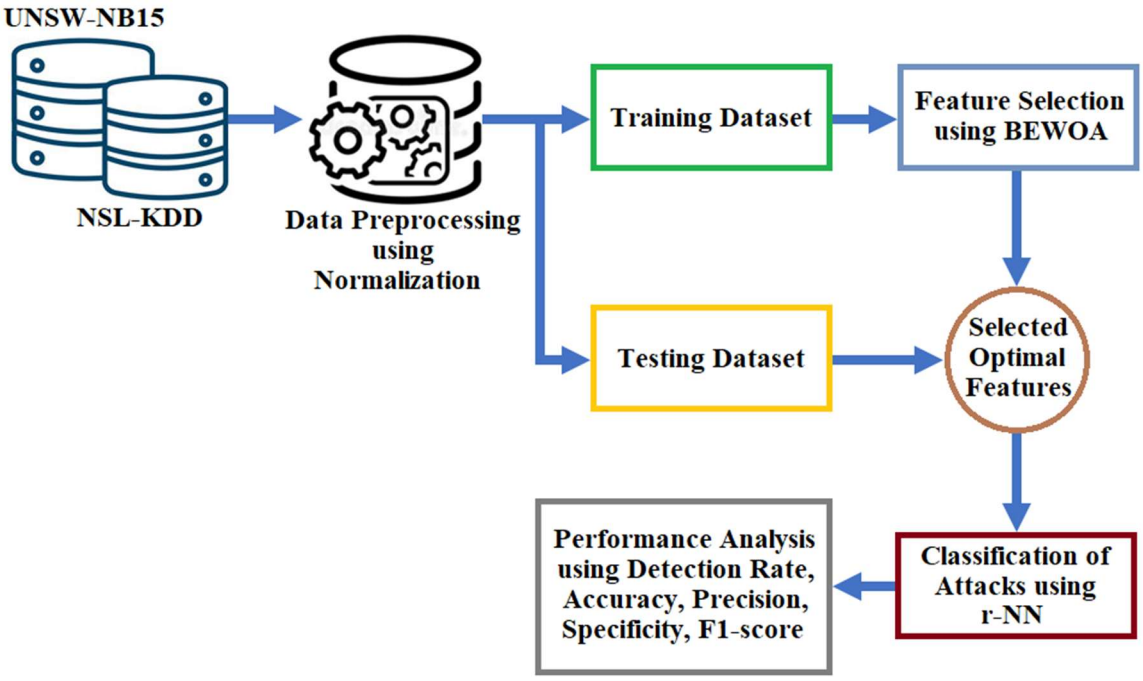

Figure 3: Proposed Model Workflow

3.2. NSL-KDD Dataset

One of the datasets that was utilized for the purpose of evaluation in this work was this dataset. It was the upgrade made to KDDcup99 dataset by eliminating the duplicate and redundant data from the origin dataset. It has 41 attributes to define each action in the IoT framework and the targets, and it can convert between the five different types of Normal, DoS, Probe, U2R, and R2L. The primary goal of the advantages provided by NSL-KDD is to reduce the degree of difficulty posed by the origin data set, KDD99. Nonetheless, it suffers from the similar issues when it comes to the modelling of real networks. The NSL-KDD data set was commonly utilized in the intrusions detection as well as various fields that are closely connected [16].

TABLE 1: DISTRIBUTION OF NSL-KDD

Traffics	Samples to Train	Samples to Test
R2L	995	2421
DoS	45927	7458
Normal	67343	9711
Probe	11656	2754
U2R	52	200
Total	125973	22544

3.3. UNSW NB-15 Dataset

This data collection was originated in the Cyber-Range Laboratory of the Australian Centre for Cyber-Security. The UNSW NB-15 data collection was designed with the intention of producing patterns of usual actions and attacks. The recording of network traffic of one hundred gigabytes was made possible with the assistance of the IXIA perfect storm and the tcpdump tools. Tools such as Argus and Bro-IDS were used in order to extract 49 features from the dataset. This was done for the aim of analyzing the dataset. The execution of the simulations led to the development of four CSV files, each of which included either normal or aberrant records depending on the particular file. The training set contains a total of 175,341 samples, but the testing set only contains 82,332 total samples [17]. These samples comprise attacks and normal types, as indicated in table II.

Table 2: UNSW-NB-15 Dataset Distribution

Traffic Labels	Samples to Train	Samples to Test
Worms	130	44
Backdoors	1746	583
Shellcode	1133	378
Analysis	2000	677
DoS	12264	4089
Reconnaissance	10491	3496
Generic	40000	18871
Fuzzers	18184	6062
Exploits	33393	11132
Normal	56000	37000
Total	175341	82322

3.4. Data Preprocessing

The stage of "data preprocessing," which comprises preparing and manipulating the raw dataset, is an essential part of the process. The dataset is first preprocessed in order to get it ready for the analysis step. The analysis of the data is done in order to learn about the characteristics of the data from the dataset and to get rid of any characteristics that are not required. In most cases, the initial network datasets contain missing attribute information, information that is unnecessary, and redundant values. Because of these characteristics, it brings about a decrease in the performance and efficiency of classification, which manifests as high error outputs, misclassification labels, false positives, and low performance in detection. As a result, performing pre-processing on the datasets is an absolute necessity to increase the performance of the classifier by more effectively training the models. In this case, the normalization method changes the original qualities into normal values that have a variance of one and a mean of zero, as the following given equation.

$$N_i = \frac{A_i - M_i}{SD} \quad (1)$$

In this context, N_i refers to the normalization distribution of n different attributes, A_i stands for the actual value, M_i represents the mean value, and SD stands for the standard deviation. The following equation was utilized to calculate the M_i and SD:

$$M_i = \frac{1}{n}\sum_{i=1}^{n} A_i \cdot SD \quad (2)$$

$$SD = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(A_i - M_i)} \quad (3)$$

The datasets have been preprocessed and used for future procedures, hence enhancing the overall classification performance and efficiency [18]. The mean and standard deviation were used as the basis for the processing.

3.5. Feature Selection using BEWOA

The method of feature selection decreases the risk of the model being overfit, speeds up the process of training the model, and makes the model less prone to test error. The BEWOA features selection methodology was utilized in this work for selecting the most useful attributes from the datasets that have been suggested. In general, the classification of a dataset has dimensions of the form $S_N \times F_N$, where S_N is the total number of samples and F_N was the number of features. The basic aim of the feature selection algorithm was to choose the subset S_S from the overall number of features (F_N), where the dimension of S_S was less than F_N. This was accomplished by comparing the dimensions of F_N and S_S. The following objective function may be applied in order to accomplish the collection of features that has been specified:

$$OBF = \lambda \times \gamma s + (1 - \lambda) \times \left(\frac{|S_S|}{F_N}\right) \quad (4)$$

In this case, γs denotes the classification error that occurred when utilizing S_S, $|S_S|$ was used to represent the features that were selected, and λ was utilized to maintain a healthy equilibrium between $\left(\frac{|S_S|}{F_N}\right)$ and γs.

The bit vector that has D components is used to represent the feature space in the process of features selection, which was a binary optimization issue. This bit vector is used for mapping selected/unselected features with values of 0 and 1. The BEWOA algorithm is proposed as a method for selecting useful features from the datasets used for intrusion detection in this research. BEWOA is the binary version of EWOA, and it was developed by utilizing the U-shaped transfer functions (TF) in order to discover useful attributes from the datasets that were provided.

The EWOA is superior to the classical WOA in terms of performance since it incorporates a pooling mechanism as well as three effective search approaches. These approaches are the migrating, the preferential selection, and the enhanced encircling the prey. The pooling method preserves the genetic variety of the populations through crossing the least desirable solution from all the iterations with the solutions that shows promise. The pooling mechanism was the crossover operators that combines the poorest possible solutions with the most optimal possible solutions in order to promote diversity. As the pool size reaches its maximum capacity, an existing member of the pool is removed and replaced with a new solution. The migratory search technique uses equation (5) to randomly divide up a section of the humpback whale in order to improve exploration and cover territory that has not yet been traversed. Also, it is anticipated that separating whales will improve the variety within the population, which will lead to a reduction in the amount of local optimal trapping.

$$X_i^{t+1} = X_{rnd}^t - X_{brnd}^t \quad (5)$$

Thus, X_{rnd}^t represents a position at random within the scope of the search space, and X_{brnd}^t represents a position at random within the vicinity of the best humpback whale.

Journal of Theoretical and Applied Information Technology
31st July 2023. Vol.101. No 14
© 2023 Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The canonical WOA's search-for-prey approach receives a boost in its capacity to explore thanks to the preferential selection strategy's implementation. Because the preferential picking search approach is intended to enhance the WOA's capability for exploration, it necessitates a sizable step size in order to unearth a wide solutions variety by dispersing the whales across search region's various parts. As a result, this tactic makes use of the heavy-tailed Cauchy distributions, characterized by a very high possibility of yielding values that are more extreme. The encircling prey approach that is utilized in the WOA is improved by the utilization of equation (6), in which D'^t was determined through the utilization of equation (7), and P_{rnd}^t was chosen at random from the matrix pool.

$$X_i^{t+1} = X_{best}^t - A_i^t \times D'^t \quad (6)$$

$$D'^t = |C_i^t \times X_{best}^t - P_{rnd}^t| \quad (7)$$

Thus, A_i^t denotes the coefficient vector, X_{best}^t represents the location of the best humpback whale, and C_i^t identifies the current location of the i^{th} whale [19].

The following is a description of the step-by-step process that is used in the BEWOA.

Step-1.) Initialization: Let the matrix $B^t = (B_1^t, B_2^t, \dots, B_N^t)$ be the binary matrices with N whale populations, and let the vectors $B_i^t = \left(b_{i,1}^t, b_{i,2}^t, \dots, b_{i,D}^t\right)$ be the bit vectors having D components for mapping selected/unselected attributes with values of 0 and 1. In the initial iteration, all the members $b_{i,j}^t$ of the vectors B_i^t was initialized by utilizing Eq. (8), where $rand$ was a random number that falls between the intervals of 0 and 1.

$$b_{i,j}^t = \begin{cases} 1 & rand \geq 0.5 \\ 0 & rand < 0.5 \end{cases}, \; i = 1,2,\dots,N \; and \; j = 1,2,\dots,D \quad (8)$$

Step-2.) Movements: To begin, the humpback whale subset P was chosen at random to be segmented and have its positions updated utilizing the migrating search approach outlined in equation (5). After that, Eq. (9) is used to relocate the remainder of them.

$$X^t = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,D} \\ x_{2,1} & x_{2,2} & \dots & x_{2,D} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N,1} & x_{N,2} & \dots & x_{N,D} \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 0 & \cdots & 1 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & \cdots & 1 \end{bmatrix} \quad (9)$$

Step-3.) TF: The U-shaped TF was utilized for mapping the constant search spaces to the binary one utilizing the equation (10), here β and α defines the width and slope of the basins in the TF and the operators $|\cdot|$ return the total values of all the elements of x_{ij}^{β}. Altering the values of the control parameter β results in a change in the breadth of the basin, which determines whether the operation is exploratory or exploitative. At the end, the component of b_{ij}^t is determined by utilizing Equation (11), where the value of 1 indicates that the feature was useful and chosen, whereas the value of 0 indicates that the feature was redundant and is not chosen.

$$U(x_{ij}^t) = \alpha \times |x_{ij}^{\beta}| \quad (10)$$

$$b_{ij}^t = \begin{cases} 1 & U(x_{ij}^t) \geq rand(0,1) \\ 0 & U(x_{ij}^t) \geq rand(0,1) \end{cases} \quad (11)$$

Step-4.) Evaluation of Fitness: The challenge of features selection was presented as the optimization issue of multi-objective with the goals of increasing the accuracy of classification while decreasing the total count of features that are chosen. Recurrently, the newer positions of all the humpback whales were analyzed by applying the objective functionality F_i^t, which is given in Equation (12). In this equation, $\alpha \in [0,1]$ and $\beta = (1-\alpha)$ were the weight factors that show the effect of quality of classification and reduction rate of feature, respectively. The overall count of features chosen by the i^{th} humpback was denoted by the notation SF_i^t whereas the overall count of features included in the initial data set was denoted by TF. The k-NN approach was utilized for computing the classification errors (EC_i^t) based on various wrapper-based features selection approaches because of its straightforward learning procedure, straightforward implementations, and minimum computation cost. This is because the k-NN classifier has a simple learning process. Concerning the related studies, the EC_i^t was calculated according to the mean square errors utilizing equation (13), here K was the overall count of folds in the K-folds CV process, y_p and y_a were the predicted and actual value in the test sets. N_{SP} was the total count of samples. In addition to that, the entropy measure is used in the computation of the EC_i^t value.

$$F_i^t = \alpha \times EC_i^t + \beta \times \left(\frac{SF_i^t}{TF}\right) \quad (12)$$

$$EC_i^t = \frac{1}{K}\sum_{fol}^{K} \quad \frac{1}{N_{SP}}\sum_{S=1}^{N_{SP}}\sqrt{(y_a - y_p)^2} \tag{13}$$

Step-5.) Updating and evaluating the global best position of whales: When determining each whale's new fitness value, compare it to its present value, and then choose the one that is superior as the new fitness value. In addition, deciding the best global value.

Step-6.) Updating the pool of matrix: Using the pooling technique, this matrix will be updated.

Step-7.) Termination of the algorithm: If the termination requirement is met, the algorithm will be terminated; otherwise, the search process will continue with step 2.

3.6. Classification using Random Nearest Neighbor

The k-NN algorithm is a powerful method that can classify unknown points by making use of the concept of majority vote. This method classified points based on the distance to their nearest neighbours. The r-NN method was a novel approach that was developed by fusing the k-NN algorithm with majority voting. This approach works well with categorical data, which might be difficult to calculate due to their nature. This approach is an ensemble of many outputs from a single methodology and takes into consideration the majority vote of all the points of classes to determine which result should be considered the primary output. By using this strategy, the highest accuracy rate will be reduced, along with the bias error. To classify the attacks using r-NN, the same method as k-NN was employed. Hence, the step-by-step method was applied from 1 through 7 with respect to Eq. (14) till Eq. (16) [20-23].

The following is a detailed description of each step involved in the r-NN method:

Step-1: Select R as the count of the nearest neighbour, and T as the set for the training samples.

Step-2: Let $Z = (x^1, y^1)$

Step-3: Using the Euclidean distance, compute the distance that exists between Z and each sample when $(x, y) \in T$.

$$ED = \sqrt{(x_i - x_i^1)^2} \tag{14}$$

Step-4: Choose the r-set that is closer to Z.

Step-5: For the classifications that we acquire,

$$y^1 = \arg\max \sum_{(x_i,y_i)\in T_z}^{\infty} I(V = Y^i) \tag{15}$$

In this context, the class denoted by y^1 represents a single point.

Step-6: Using the equation from step 5, we acquire,

$$\hat{y}_i = \arg\max \sum_{i\in Z}^{\infty} \left[y_{m,r_j}\right] \tag{16}$$

Here, j represents the value of R, m representing the number of samples, and r_j refers to the value of K.

Step-7: Repetition of steps 3-6 until all of the points have been classified.

Equation (16) can be used to r-NN in the following way:

$$\begin{gathered}\arg\max\left[y_{1,k_1}, y_{1,k_2}, \dots, y_{1,k_m}\right] = \hat{y}_1 \\ \arg\max\left[y_{2,k_1}, y_{2,k_2}, \dots, y_{2,k_m}\right] = \hat{y}_2 \\ \vdots \\ \arg\max\left[y_{\infty,k_1}, y_{\infty,k_2}, \dots, y_{\infty,k_m}\right] = \hat{y}_\infty\end{gathered} \tag{17}$$

There are processes for classifying the attacks according to the nearest neighbours with considering a single neighbour starting at step 1 to step 6. When dealing with categorical information, the idea of a majority vote is applied, and the values $\hat{y}_1, \hat{y}_2, \dots, \hat{y}_\infty$ indicate the classifications that result from each attack. It demonstrates how r-NN operates by using equation (17), and it classified each attack using different values of k. After the optimal feature subset in the training data set had been chosen, and it had been established that the number of false positives in the intrusion detection process generated by using this dataset was kept to a minimum, a comparable subset of feature selection patterns could be utilized to the test data set for classifying the attacks using the r-NN classifier.

4. RESULTS AND DISCUSSION

The performance analysis of the proposed BEWOA-rNN model using the datasets that were utilized for this research as well as the discussion of the results were presented in this section. This proposed model used MATLAB 2018a for its implementation and evaluation, performed on a laptop equipped with a CPU running at 2.80 GHz and 8 GB of RAM. The proposed research model was evaluated based on the parameters of the results like accuracy, detection rate, f-measure, specificity, and precision.

4.1. Result Parameters

To calculate the efficiency of the research model for detecting the intrusions in IoT environments, various performance measures are evaluated. Accuracy, specificity, detection rate, f-measure and precision are the performance metrics and all of these are based on the confusion matrix described in table 3.

Journal of Theoretical and Applied Information Technology
31st July 2023. Vol.101. No 14
© 2023 Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The performance metrics are evaluated based on the formulation of the confusion matrix. As shown in table III, TP indicates the true positive, FN indicates the false negative, FP indicates the false positive and TN indicates the true negative.

$$Accuracy = \frac{TP+TN}{FN+TP+FP+TN} \quad (18)$$

$$Detection\ rate = \frac{TP}{TP} \quad (19)$$

$$Specificity = \frac{TN}{TN+FN} \quad (20)$$

$$Fmeasure = 2 \times \frac{Precision \times Recall}{Precision+Recal} \quad (21)$$

$$Precision = \frac{TP}{TP+FP} \quad (22)$$

Table 3: Confusion Matrix Formulation

	Predicted Label	
Actual Label	**Positive**	**Negative**
Positive	TP	FN
Negative	FP	TN

Using the proposed BEWOA algorithm, from the NSL-KDD dataset, feature numbers (7, 15, 18, 20, 21, and 24) were not selected and the remaining 35 features out of 41 features were selected. From the UNSW-NB-15 dataset, out of 47 features only 12 features (1, 3, 4, 5, 7, 9, 39, 42, 43, 44, 45, and 47) were selected using the BEWOA algorithm.

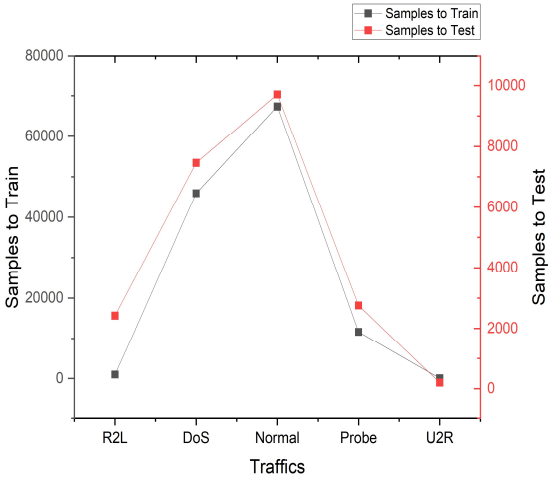

Figure 4: Traffic Distributions of NSL-KDD Dataset

Figures 4 and 5 represents the traffic distributions of datasets used for this research to experiment and evaluate the performance of the research model.

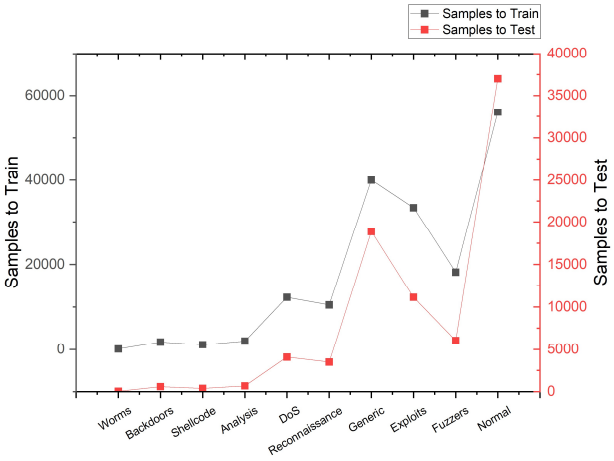

Figure 5: Traffic Distributions of UNSW-NB-15 Dataset

The datasets were divided into 80% to train and 20% to test the BEWOA-rNN model. The confusion matrix plot yielded by the proposed model for the NSL-KDD was represented in figure 6 and for the UNSW-NB-15 the confusion matrix plot was depicted in figure 7.

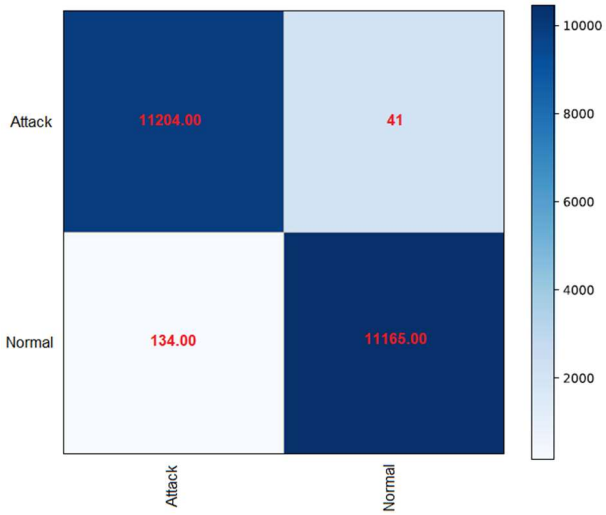

Figure 6: NSL-KDD Dataset Confusion Matrix

Journal of Theoretical and Applied Information Technology
31st July 2023. Vol.101. No 14
© 2023 Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

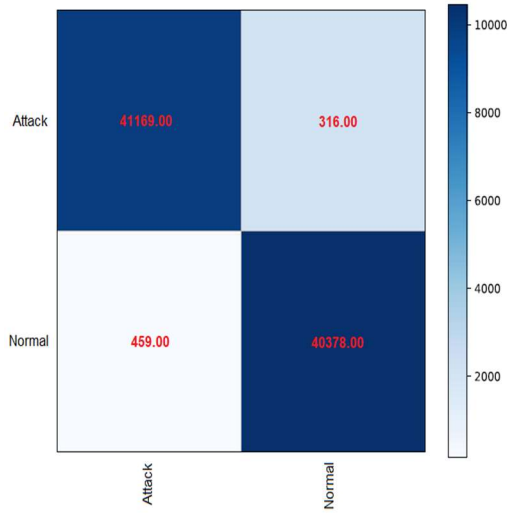

Figure 7: UNSW-NB-15 Dataset Confusion Matrix

Table 4: Results of the BEWOA-rNN model on Training Set

Metric (%)	NSL-KDD	UNSW-NB-15
Accuracy	99.55	99.41
Detection Rate	99.63	99.53
Specificity	97.34	94.69
Precision	99.90	99.86
F-measure	99.76	99.69

Based on the training set and testing set, the performance analysis was performed individually with both the datasets in terms of the result parameters. Table IV represents the results of the proposed model evaluated using the training set extracted from the proposed datasets.

As shown in the table, the research model obtained effective performance in all the measured result parameters. The BEWOA-rNN model achieved 99.55% accuracy for NSL-KDD and 99.41% accuracy for UNSW-NB-15. The research model has obtained higher accuracy for NSL-KDD compared to the UNSW-NB-15. The detection rate or recall or sensitivity of the BEWOA-rNN model using the NSL-KDD was 99.63% and 99.53% for UNSW-NB-15. The specificity rate obtained by the proposed model using the NSL-KDD was 97.34% and 94.69% for the UNSW-NB-15. The specificity rate difference among the datasets was 2.65%, in which the model obtained higher specificity rate using NSL-KDD dataset. The precision rate was 99.90% using NSL-KDD and 99.86% using the UNSW-NB-15. The f-measure rate of the research model obtained utilizing the NSL-KDD was 99.76% and 99.69% using the UNDW-NB-15 dataset. Overall, the performance analysis of the BEWOA-rNN model using the training datasets was higher in using the NSL-KDD. The graphical plot for the performances analysis obtained utilizing the training sets was represented in figure 8.

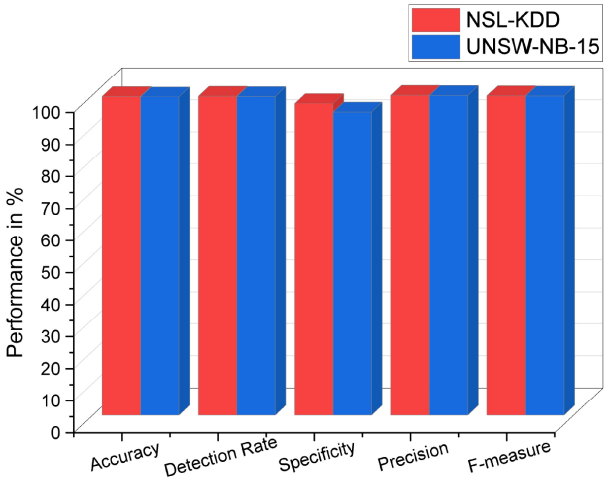

Figure 8: Proposed Model's Performance Analysis on Training Set

Table V represents the performances evaluation of the research model using the test datasets individually. The BEWOA-rNN model achieved 99.22% accuracy for NSL-KDD and 99.06% accuracy for UNSW-NB-15. The research model has obtained higher accuracy for NSL-KDD compared to the UNSW-NB-15. The detection rate of the BEWOA-rNN model using the NSL-KDD was 98.82% and 98.90% for UNSW-NB-15. The specificity rate obtained by the proposed model using NSL-KDD was 99.63% and 99.22% for the UNSW-NB-15. The precision rate was 99.64% using NSL-KDD and 99.24% using the UNSW-NB-15. The f-measure rate of the research model obtained utilizing the NSL-KDD was 99.23% and 99.07% using the UNDW-NB-15 dataset. Overall, the performance analysis of the BEWOA-rNN model using the test datasets was higher in using the NSL-KDD. The graphical plot for the results obtained utilizing the test sets was represented in figure 9.

Table 5: Results of the BEWOA-rNN model on Test Set

Metric (%)	NSL-KDD	UNSW-NB-15
Accuracy	99.22	99.06
Detection Rate	98.82	98.90
Specificity	99.63	99.22
Precision	99.64	99.24
F-measure	99.23	99.07

Journal of Theoretical and Applied Information Technology
31st July 2023. Vol.101. No 14
© 2023 Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

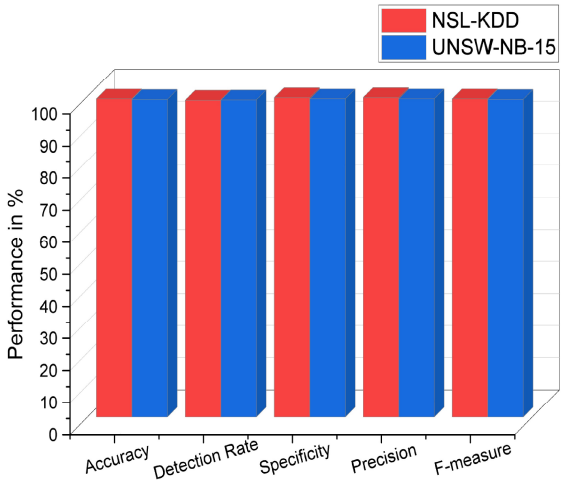

Figure 9: Proposed Model's Performance Analysis on Training Set

The research model's experiment analysis was compared with the existing models for proper validation. The existing models are derived from the discussed works analyzed in the background survey. The existing models are support vector machines (SVM), logistic regression (LR), Artificial Bee Colony with Artificial Fish Swarm (ABC-AFS), Correlation-based feature selection with Bat algorithm and C4.5 (CFS-BA-C4.5), and ForestPA (CFS-BA-ForestPA) and Intrusion Detection Tree (IntruDTree). Table VI represents the performance analysis comparison of BEWOA-rNN model with other models.

Table 6: Performances Comparison of the BEWOA-rNN

Models	Accuracy (%)	Detection Rate (%)	Precision (%)	F-measure (%)
LR [6]	98.30	98.00	98.00	98.00
SVM [6]	98.20	98.00	98.00	98.00
ABC-AFS [8]	97.60	98.60	NA	NA
CFS-BA-C4.5 [11]	98.80	98.80	98.70	98.80
CFS-BA-ForestPA [11]	98.70	98.70	98.90	98.80
IntruDTree [12]	98.00	98.00	98.00	98.00
BEWOA-rNN (NSL-KDD)	99.22	99.82	99.64	99.23
BEWOA-rNN (UNSW-NB-15)	99.06	98.90	99.24	99.07

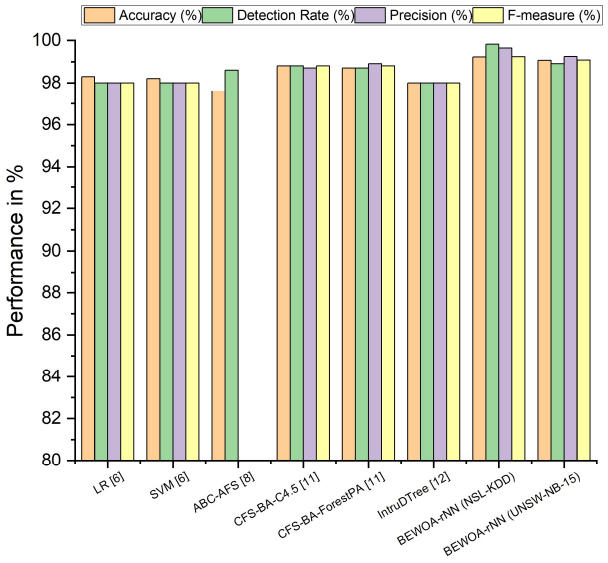

Figure 10: Overall Comparison of Performance Analysis

As represented in table VI, the research model's performances were compared with the other models for validation. The result parameters like accuracy, detection rate, precision, and f-measure scores of the BEWOA-rNN model evaluated using NSL-KDD and UNSW-NB-15 datasets were utilized in this comparison. The highest accuracy score was obtained by the proposed model with 99.22%, in which the proposed model was 0.42% to 1.62% better than the other models. The best detection rate was achieved by the proposed model with 99.82%, where the proposed model was 1.02% to 1.82% higher than the compared models. The precision score of the proposed model was 99.64%, in which the proposed model obtained higher precision score with 0.74% to 1.64% differences among the compared models. The f-measure score of the proposed model was 99.23%, where the proposed model's f-measure score was 0.43% to 1.23% better than the other models in this comparison. Figure 9 represents the comparison of the research model's performances with compared models. Overall, the proposed BEWOA-rNN model has achieved better results in all the parameters with efficient performance and outperformed the other compared models.

The limitations of the previous works compared to the BEWOA-rNN model:

- The work in [6] conducted in a simulated environment raises concerns about the generalizability of the model to real-time

data. The accuracy of 99.4% achieved in the simulated environment may not necessarily translate to real-world scenarios.

- While the EASH system in [7] achieved 85% accuracy, there is room for improvement through the incorporation of dimensionality reduction techniques or feature selection methods to enhance accuracy further.
- The hybrid IDS model based on artificial bee colony and artificial fish swarm algorithms in [8] could be improved in terms of computational difficulty and the amount of time required for processing. These factors may limit its scalability and practical implementation.
- The BAT model in [9] for traffic anomaly detection achieved an accuracy of 84.25% for time series data, indicating room for improvement in terms of accuracy.
- Although the CART algorithm [10] used in detecting Botnet attacks demonstrated high detection rates with minimal processing load, the performance and robustness of the model in handling various types of attacks and real-world scenarios are not specified.
- While the framework for intrusion detection based on feature selection and ensembled learning models in [11] achieved better detection rates and accuracy, the limitations of the model in terms of computational complexity and scalability are not addressed.
- The IntruDTree model in [12] focused on identifying abnormalities and attacks based on patterns in the security dataset. However, the effectiveness and performance of the model on unseen test cases are not explicitly mentioned.
- Although the two-stage hybrid method for IoT intrusion detection in [13] showed promising results compared to other classifiers, the specific limitations and trade-offs of each classifier in terms of precision, computational cost, and false positives are not discussed.
- The lightweight IDS models for 6LoWPAN networks in [14] achieved varying detection rates for different attack scenarios, but the performance and limitations of these models in terms of scalability and handling more complex attacks are not clearly addressed.
- The hybrid approach in [15] that combines data-level techniques and feature-level approaches shows the potential in enhancing the detection rate of the IDS system. However, the impact on computational time and the scalability of the model with larger datasets are not explicitly discussed.

5. CONCLUSION

In this research, a novel hybrid IDS model was proposed to detect and classify the attacks in the IoT environment. The proposed model was a combination of a metaheuristic algorithm and a machine learning technique. The metaheuristic algorithm called BEWOA was used as the feature selection technique and the machine learning algorithm called r-NN was used for the classification. The proposed model was evaluated with two datasets such as NSL-KDD and UNSW-NB-15. The workflow of the model includes the preprocessing, dataset splitting, feature selection and classification. The preprocessing was performed using the normalization technique for data standardization and data cleaning. After preprocessing, the features from the datasets were selected using the BEWOA algorithm for improving the classification performance of the model. Using the BEWOA algorithm, 35 features out of 41 from the NSL-KDD and 12 features out of 47 from the UNSW-NB-15 were selected. Based on the features selected the classification was performed and the performance was measured with detection rate, accuracy, specificity, f-measure, and precision. The performance evaluation was measured individually for both the datasets using the BEWOA-rNN model. The model obtained 99.22% accuracy, 98.82% detection rate, 99.63% specificity, 99.64% precision, and 99.23% f-measure using the NSL-KDD dataset. The model obtained 99.06% accuracy, 98.90% detection rate, 99.22% specificity, 99.24% precision, and 99.07% f-measure using the UNSW-NB-15 dataset. The research model's performances were compared with the other models for validation and the BEWOA-rNN model outperformed the compared models on classifying the attacks.

From the authors point of view, overall, the presented research introduces a hybrid IDS model that utilizes a metaheuristic algorithm for feature selection and a machine learning algorithm for classification in the IoT environment. The model demonstrates strong performance on the evaluated datasets and shows promising results compared to other models. Based on the provided objectives, it can be concluded that the research focuses on addressing the issue of attack detection

and classification in IoT environments. The proposed BEWOA-rNN model is a novel hybrid IDS approach that combines a metaheuristic algorithm (BEWOA) for feature selection and a machine learning algorithm (r-NN) for classification. The evaluation of the proposed model using NSL-KDD and UNSW-NB-15 datasets, along with the measurement of various performance metrics, demonstrates a thorough analysis of the model's effectiveness. Comparative analysis with other models adds value to the research, highlighting the superiority of the BEWOA-rNN model in classifying attacks. However, it should be noted that the limitations of previous works are mentioned in the provided information. In conclusion, the objectives of the research aim to propose and evaluate a hybrid IDS model that leverages metaheuristics and machine learning techniques to detect and classify attacks in the IoT environment. The research provides valuable insights and performance metrics, showcasing the potential of the BEWOA-rNN model as an effective solution.

The main limitation of the proposed model can be overcome, which is the binary classification. The present model is based on binary classification and it yielded results as either the attack present or not. In future, the proposed model can be developed as a multi-class classification, which will provide the detailed results on attacks individually according to the performance.

REFERENCES

[1] M. Burhan, R. A. Rehman, B. Khan, and B. S. Kim, "IoT Elements, Layered Architectures and Security Issues: A Comprehensive Survey," *Sensors*, Vol. 18, No. 2796, 2018, pp. 1-37.

[2] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, "A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications," *IEEE Internet of Things Journal*, Vol. 4, No. 5, 2017, pp. 1125-1142.

[3] J. C. S. Sicato, S. K. Singh, S. Rathore, and J. H. Park, "A Comprehensive Analyses of Intrusion Detection System for IoT Environment," *J. Inf. Process. Syst*, Vol. 16, No. 4, 2020, pp. 975-990.

[4] N. Chaabouni, M. Mosbah, A. Zemmari, C. Sauvignac, and P. Faruki, "Network Intrusion Detection for IoT Security based on Learning Techniques," *IEEE Commun. Surv. Tutor.*, Vol. 21, No. 3, 2019, pp. 2671-2701.

[5] S. Choudhary and N. Kesswani, "A Survey: Intrusion Detection Techniques for Internet of Things," *Int. J. Inf. Secur. Priv.*, Vol. 13, No. 1, 2019, pp. 86-105.

[6] M. Hasan, Md. M. Islam, Md. I. I. Zarif, and M. M. A. Hashem, "Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches," *Internet of Things*, Vol. 7, No. 100059, 2019, pp. 1-14.

[7] D. J. Atul, R. Kamalraj, G. Ramesh, K. S. Sankaran, S. Sharma, and S. Khasim, "A machine learning based IoT for providing an intrusion detection system for security," Microprocess. *Microsyst.*, Vol. 82, No. 103741, 2021, pp. 1-10.

[8] V. Hajisalem and S. Babaie, "A hybrid intrusion detection system based on ABC-AFS algorithm for misuse and anomaly detection," *Comput. Netw.*, Vol. 136, 2018, pp. 37–50.

[9] T. Su, H. Sun, J. Zhu, S. Wang, and Y. Li, "BAT: Deep Learning Methods on Network Intrusion Detection Using NSL-KDD Dataset," *IEEE Access*, Vol. 8, 2020, pp. 29575-29585.

[10] C. S. Htwe, Y. M. Thant, and M. M. S. Thwin, "Botnets Attack Detection Using Machine Learning Approach for IoT Environment," *6th Annual International Conference on Network and Information Systems for Computers*, Guiyang, China, Vol. 1646, 2020, pp. 1-7.

[11] Y. Zhou, G. Cheng, S. Jiang, and M. Dai, "Building an efficient intrusion detection system based on feature selection and ensemble classifier," *Comput. Netw.*, Vol. 174, No. 107247, 2020, pp. 1-17.

[12] I. H. Sarker, Y. B. Abushark, F. Alsolami, and A. I. Khan, "IntruDTree: A Machine Learning Based Cyber Security Intrusion Detection Model," *Symmetry*, Vol. 12, No. 754, 2020, pp. 1-15.

[13] T. Saba, T. Sadad, A. Rehman, Z. Mehmood and Q. Javaid, "Intrusion Detection System Through Advance Machine Learning for the Internet of Things Networks," *IT Professional*, Vol. 23, No. 2, 2021, pp. 58-64.

[14] P. Shukla, "ML-IDS: A Machine Learning Approach to Detect Wormhole Attacks in Internet of Things," *2017 Intelligent System Conferences (IntelliSys)*, London, UK, 2017, pp. 234-240.

[15] U. R. Salunkhe and S. N. Mali, "Security Enrichment in Intrusion Detection System Using Classifier Ensemble," *J. Electr. Comput.*

Eng., Vol. 2017, Article ID 1794849, 2017, pp. 1-6.

[16] K. Albulayhi, Q. A. Al-Haija, S. A. Alsuhibany, A. A. Jillepalli, M. Ashrafuzzaman, and F. T. Sheldon, "IoT Intrusion Detection Using Machine Learning with a Novel High Performing Feature Selection Method," *Appl. Sci*., Vol. 12, No. 5015, 2022, pp. 1-30.

[17] I. Tareq, B. M. Elbagoury, S. El-Regaily, and E. S. M. El-Horbaty, "Analysis of ToN IoT, UNW-NB15, and Edge-IIoT Datasets Using DL in Cybersecurity for IoT," *Appl. Sci*., Vol. 12, No. 9572, 2022, pp. 1-26.

[18] D. Hemanand, G. Vinoda Reddy, S. Sathees Babu, Kavitha Rani Balmuri, T. Chitra, and S. Gopalakrishnan, "An Intelligent Intrusion Detection and Classification System using CSGO-LSVM Model for Wireless Sensor Networks (WSNs)," *Int. J. Intell. Syst. Appl. Eng*., Vol. 10, No. 3, 2022, pp. 285-293.

[19] M. H. N. Shahraki, H. Zamani, and S. Mirjalili, "Enhanced whale optimization algorithm for medical feature selection: A COVID-19 case study," *Comput. Biol. Med*., Vol. 148, No. 105858, 2022, pp. 1-30.

[20] A. H. B. Kamarulzalis and M. A. A. B. Abdullah, "An Improvement Algorithm for Iris Classification by using Linear Support Vector Machine (LSVM), k-Nearest Neighbours (k-NN) and Random Nearest Neighbours (RNN)," *J. Math. Comput. Sci*., Vol. 5, No. 1, 2018, pp. 32-38.

[21] D. Sathiya and S. Sheeja, "Data Delivery and Node Positioned Learning Automaton in Mobile Ad Hoc Networks," *J. Comput. Sci. Intell. Technol.,* Vol. 3, No. 2, 2022, pp. 1-14. https://doi.org/10.53409/MNAA/JCSIT/e202203020114

[22] S. S. Lutfi and M. L. Ahmed, "A Novel Intrusion Detection System in WSN using Hybrid Neuro-Fuzzy Filter with Ant Colony Algorithm," *J. Comput. Sci. Intell. Technol.,* Vol. 1, No. 1, 2020, pp. 1-8. https://doi.org/10.53409/mnaa.jcsit1101

[23] R. Khilar et al., "Artificial Intelligence-Based Security Protocols to Resist Attacks in Internet of Things," *Wireless Commun. Mobile Comput.*, Vol. 2022, No. 1440538, 2022, pp. 1-10. https://doi.org/10.1155/2022/1440538