Received: September 13, 2024. Revised: October 30, 2024. 171

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Network Intrusion Detection System Using Reptile Search with Whale Optimization Algorithm and Multi Head Attention Long Short Term-Memory in IoT

Vishwanath Digambar Chavan[1]* **Pratibha Chidanand Kaladeep Yalagi[1]**

[1]*Department of Computer Science and Engineering,*
Walchand Institute of Technology, Solapur, Maharashtra, India
* Corresponding author's Email: vdc.chavan@gmail.com

Abstract: The Internet of Things (IoT) introduces new technological advancements for the development of diverse significant applications. To address the need for protection against attacks, fraud and network intrusions, the Intrusion Detection System (IDS) has become a crucial component within organizations. However, due to the limited resources of IoT devices, classifying attacks and training the model on large datasets consumes more time. In this research, the proposed approach combines the Reptile Search Algorithm (RSA) and Whale Optimization Algorithm (WOA) with Multi-head attention with Long Short-Term Memory (MHA-LSTM) for effective and accurate IDS classification. Initially, IDS is obtained from CICIDS2017, CICIDS2018 and NSL-KDD. Pre-processing methods like Standard Scaling and Label Encoding are involved in transforming numerical values, segmenting features and adjusting them using mean and standard deviation to reduce sensitivity. The RSA with WOA is involved in enhancing intrusion detection by selecting relevant features and optimizing the detection process efficiently to solve complex optimization problems. The classification combination of MHA-LSTM allows the model to scale effectively to large datasets and maximum complex tasks without compromising the performance and accuracy. The proposed WOA-RSA – MHALSTM technique is evaluated on CICIDS 2017, CICIDS 2018 and NSL-KDD datasets, achieving higher accuracies of 99.997% on CICIDS2017, 99.99% on CICIDS2018 and 99.99% on NSL-KDD datasets, which is more effective than Deep Neural Network (DNN) and LSTM.

Keywords: Deep neural network, Long short-term memory, Intrusion detection system, Reptile search algorithm, Whale optimization algorithm.

1. Introduction

Internet of Things (IoT) networks comprise a diverse array of smart devices in the environment that collect, process and transmit data [1]. The aim is to digitize every physical object, connecting billions of devices in IoT, each embedded with sensors and other technologies that generate huge data [2]. Incidents involving IoT attacks have significantly increased over the past year, both in terms of frequency and complexity. The network comprises many interconnected devices, including cameras, temperature sensors, smart TVs, and wireless printers, all requiring network connectivity [3]. The utilization of research, innovation, and cybercriminal demand has stimulated the interest of potential investors. However, it has also created opportunities for cybercriminals to exploit weaknesses and vulnerabilities in IoT devices [4]. Therefore, a DL approach implemented for IDS must have a lightweight architecture, while the network compression decreases the amount of required processing resources of the trained network, reducing the dimensions of data [5]. These systems identify malicious attacks and isolate them from normal traffic, necessitating the extensive usage of IoT for handling security challenges [6]. CICIDS2018 and NSL-KDD datasets are generated, corresponding to data features from the network data flows that are ideally labelled with attacks or benign classes to

allow for DL techniques [7]. It is designed to provide security threats and protection in operational infrastructures, aiming to preserve the principles of information system security, including confidentiality and integrity [8].

A serious security issue for intrusion detection systems is the facing of malicious software such as U2R, Denial of Service attacks (DDoS), Probes, and R2L, leading to network security breaches and serious faults [9]. In this research, an IDS model for IoT security is designed and validated with RSA and WOA combined to enhance feature selection. There are consider various optimization such as Dollmaker Optimization Algorithm (DOA) approach. This aims to reduce features and improve performance in terms of accuracy, precision, and data preparation for the training phase [10]. However, upgradability is based on cognitive radio technology, which has a significant importance in recognizing the environment for devices [11]. The goal of this research is to improve the accuracy of intrusion detection and combine feature selection and classification to enhance the IDS system [12]. However, it is challenging to detect IoT attacks because the IoT traffic originates from diverse sources, making it challenging to distinguish attack traffic from normal traffic [13]. As IoT devices are heterogeneous and follow different protocols, various security measures need to be followed due to their seamless nature [14]. The objective of host-based IDS systems is to develop lightweight intrusion prevention software tailored for Python systems. This includes a management console that introduces network monitoring capabilities in software design security, enabling the early detection of network attacks [15]. From the overall analysis, it is seen that the existing techniques have limitations, due to the limited resources of IoT devices, where classifying attacks and training the model on large datasets consumes more time. In this research, the goal of the process is to explore the search space of possible solutions, refine attacks, ensure adequate execution time, and achieve an efficient convergence rate. The feature selection using RSA with WOA is performed relevant feature and balance exploration and exploration in feature space.

The main contributions of the research are as written below:

- RSA combined with WOA is employed for feature selection to efficiently select relevant features for intrusion detection. It explores the search space of potential solutions, enhances attack detection accuracy, ensures adequacy, and achieves efficient convergence rates.
- WOA and RSA are sequentially performed in the feature selection process by extracting related features and determining the best features for intrusion detection.
- The intrusion detection system uses MHA-LSTM to effectively classify data as normal or attacks. This model is adept at handling large datasets and complex tasks while maintaining a superior performance and accuracy.

The paper is organized as follows: Section 2 gives an account of the related work that summarizes IDS using DL techniques, Section 3 introduces the proposed method utilized by WOA-RSA-based MHA-LSTM, while Section 4 discusses the result and comparative analysis, and finally, the conclusion of this research is given in Section 5.

2. Related work

A large number of researchers conduct studies on network intrusion detection, analysing network traffic to detect normal and attack behaviours using DL algorithms, which are widely used in designing IDS. The related works of IDS in IoT are provided in this section, along with their advantages and limitations.

Dahou [16] introduced IDS for IoT, integrating DL and an enhanced Reptile Search Algorithm (RSA) model that combined DL and metaheuristic optimization for feature extraction and selection. By utilizing a Convolutional Neural Network (CNN) as a feature extractor and RSA for feature selection, the IDS enhanced performance by selecting the most significant features from the extracted features. However, this approach required increased computational resources for packet investigation during attack detection.

Selvapandian and Santhosh [17] introduced a LeNet architecture for IDS that was implemented in the IoT multi-cloud environment. This DL-based IDS aimed to overcome the limitations of neural network-based IDS. Leveraging NSL-KDD dataset, the implemented LeNet-based IDS demonstrated enhanced performance, offering a high convergence rate and ease of input computation. The method also improved detection accuracy by enhancing training efficiency. However, there was a need to enhance the effective detection of multi-class attacks in a cloud environment.

Vishwakarma and Kesswani [18] implemented a real-time IDS utilizing a Deep Neural Network (DNN) trained on 28 features. The system incorporated a pipeline with progressive components for encoding categorical information and scaling features. Real-time data was employed to train the

DNN model for predictions. The IDS was deployed on a server, enabling widespread access and integration with local networks. However, this method demanded prolonged training time, elevated computational expenses, and also encountered difficulties in acquiring labelled data for network intrusion.

Hnamte [19] implemented a DL-based technique that combines LSTM and Autoencoder (LSTM-AE) for cyberattack detection, aimed at both recognizing attacks and providing clarity on model decisions. Features were extracted, and the original input feature set was used to train the IDS. This system improved the explainability and interpretability of IoT networks while maintaining minimum computational cost and minimizing training time. However, it encountered issues with vanishing and exploding gradients, where the gradients diminished or escalated significantly during backpropagation, impacting the training effectiveness.

Keshk [20] introduced an anomaly-based IDS emerging application of DL models that necessitated the interpretation of a model explainable IDS framework in IoT networks. This framework extracted features and utilized the original set of input features for training, thereby offering both global and local explanations to IDS and enhancing the interpretability of cyber defence systems in IoT networks. Nonetheless, the intrusion detection model proved to be complex, and its back-to-back performance made it difficult to interpret.

Ameera S. Jaradat [21] presented an intrusion detection systems involve Machine Learning (ML) based Decision Tree (DT) technique. These was efficient transfer node induces a classification of decision tree in network. However, DT technique struggle to identify intrusions effectively because they consider the various attack leading to high false-negative rates.

Saleh AI Omari [22] introduced a Dollmaker Optimization Algorithm (DOA) was derived from two natural behaviours in doll making process. These was efficiently balancing the exploration and exploitation of large data and handle high dimensionality data. However, DOA involving a large number of variable and struggle to maintain population size grows then require more resource.

From the overall analysis, it is understood that the existing techniques have limitations including the limited resources of IoT devices, and more time consumption during the classification of attacks and training of the model on large datasets. In this research, the goal is to explore the search space of possible solutions, refine attacks, ensure adequate execution time, and achieve an efficient convergence rate. The feature selection using RSA with WOA performs with high accuracy for feature extraction. Here, the attack is classed as both normal and an attack using the MHA-LSTM classification method.

3. Proposed methodology

This research proposes the RSAWOA-MHALSTM technique for classification to effectively handle large datasets and intricate tasks while maintaining high performance and accuracy. The IDS utilizes datasets namely CICIDS2017, CICIDS2018 and NSL-KDD to assess the performance of the IDS system. The pre-processing techniques include Standard Scaling and Label Encoding, which distribute the features between zero and one, ensuring all features are on the same scale, also preventing extracts individual features from dominating while avoiding data leakage. Fig. 1 depicts the overview of the proposed method.

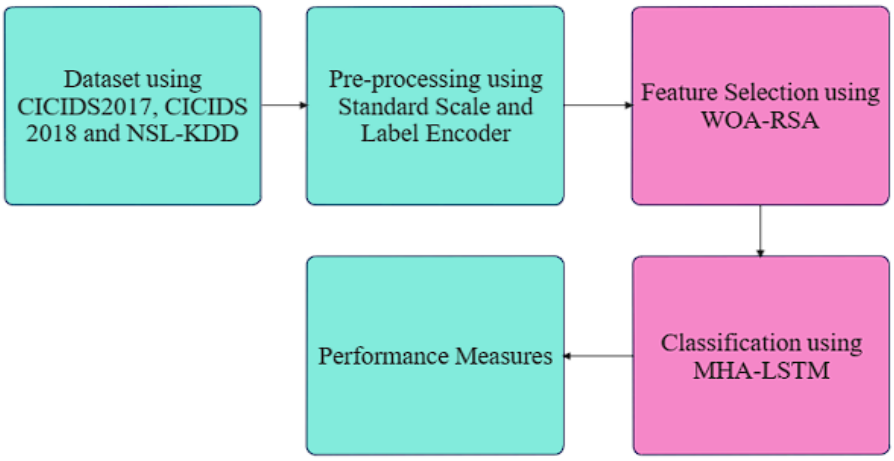

Figure. 1 Block diagram of the proposed method

3.1 Dataset

The IDS-based dataset is a multiple statistical analyses method highlighted in benchmark datasets CICIDS2017, CICIDS2018 and NSL-KDD, which pose a challenge to the accurate performance calculated with a comparison of various techniques. In order to address this, redundant and duplicate records are removed. This prevents IDS from selecting identical records across various difficulty levels, as the dataset's record count inversely relates to the difficulty level. Ensuring an adequate and balanced number of records in the datasets is imperative for achieving dependable results across different classifiers.

3.1.1. CICIDS 2017 dataset

The CICIDS-2017 dataset [23] was designed specifically for evaluating IDS in network traffic. It provides the most up to date and relevant data for testing security system. The primary for selecting this dataset is its comprehensive coverage with 80 distinct features.

3.1.2. CICIDS2018 dataset

The CICIDS2018 dataset is elaborated as Canadian Institute for Cybersecurity (CIC) combined with IDS, using the concept of profiling to build a comprehensive cybersecurity dataset [24]. It includes a wealth of data with over 80 features gathered in 6 columns, containing SourceIP, FlowID, SourcePort, DestinationIP, and Protocol.

3.1.3. NSL-KDD dataset

The NSL-KDD dataset is derived from KDDcup99 dataset, which is widely recognized as one of the most prominent intrusion datasets. However, KDDcup99 dataset suffers from issues in the training and test data of duplicate records, as well as biased classifiers. These issues have been addressed in the NSL-KDD dataset with a total record of data being 125,973, where benign and anomaly respectively are 68,353 and 58,630 samples [25]. It stands as one of the most frequently utilized datasets for evaluating IDS frameworks.

3.2 Data pre-processing

The collective datasets provided as input for preprocessing are seen as a significant task in the DL field, as they help eliminate defects in the datasets. This is the initial stage of the proposed method, designed to convert raw IoT network attack data into an effective format for further analysis. In the IDS, pre-processing techniques such as Standard Scaling and Label Encoder are combined to prevent attacks by separating the features and then using the mean and standard deviation to make them less sensitive. Standard Scaling is used to manage less to less sensitive and non-uniform values in contrast to the min-max method, which is influenced by extreme values and results in uniformly scaled data [26]. The standard scaling process involves normalizing data by subtracting mean and scaling it to have unit variance. In mathematical Eq. (1), s represents the standard deviation and μ denotes mean.

$$x_{scaler}=\frac{x-\mu_{mean}}{s_{stddiv}} \tag{1}$$

L2-standardization normalizes the dataset by collecting each row and summing the square values of each. Eq. (2) denotes L2-standardization where x represents the values of features in the dataset.

$$\|x\|_2 = (|x_1|^2 + |x_2|^2 + \cdots + |x_n|^2)^{\frac{1}{2}} \tag{2}$$

Data normalization is a crucial step in data preprocessing, especially for IDS that rely on statistical attributes extracted from the available data. Label encoding techniques handle categorical values by assigning unique numeric values to each category. Since the dataset includes feature space with multiple categories, where one-hot encoding requires more memory because it assigns either 0 or 1 for each category. Hence, the label encoder approach is preferred, converting categorical features into numeric values represented by simplified understanding through two digits, 0 and 1. Both standard scaling and label encoding techniques efficiently handle numerical values and remain unaffected by extreme values in both training and testing datasets. The pre-processed data is then made available for feature selection, facilitating the extraction of relevant information and its subsequent classification.

3.3 Feature selection

Following pre-processing techniques, the data is standardized so that each feature is ranged between zero and one, ensuring uniform scaling across all features. The goal of feature extraction is to extract relevant information by selecting features that contribute efficiently, as the expensive features lead to computational overhead and increased detection time for the IDS. The proposed method, WOA with the RSA approach initiates the process, significantly impacting IDS detection in an IoT environment.

3.3.1. Whale optimization algorithm

WOA is a metaheuristic algorithm inspired by nature, specifically designed to replicate the social behaviour of humpback whales. The goal of optimization techniques is to find the best values through exploitation and exploration. There are three different stages: encircling prey, Bubble-Net, and Search for prey [27]. Humpback whales detect the IDS position of prey and encircle them because optimal design position in search space is not always assumed to be best value. After defining the best search agent, Eqs. (3) and (4) is mathematically expressed as below.

$$\vec{D} = \left|\vec{X}.\ \vec{Y}(t) - \vec{X}(t)\right| \tag{3}$$

$$\vec{Y}(t+1) = \vec{Y}(t) - \vec{B}.\vec{E} \tag{4}$$

Where, t indicates current iteration, $\vec{B}$ and $\vec{E}$ are coefficient vectors, and X is position vector of best solution. The linearly ranged iteration encompasses both exploration and exploitation. In second stage, Bubble-Net attacking method is used for exploration stage and involves shrinking encircling and spiral updating position to achieve the decreasing value of $\vec{B}$ in level [-1, 1] to search for new position of current best agent. The spiral equation is formulated to mirror the helix-shaped movement of humpback whales, linking the position of whale prey. Eqs. (5) and (6) are presented as follows:

$$\vec{Y}(t+1) = \overrightarrow{D'}.e^{bl}.cos(2\pi l) + \vec{Y}(t) \tag{5}$$

$$\vec{Y}(t+1) = \begin{cases} \vec{Y}(t) - \vec{B}.\vec{E} & if\ p < 0.5 \\ \vec{D}.e^{bl}.\cos(2\pi l) + \vec{X}(t) & if\ p \geq 0.5 \end{cases} \tag{6}$$

Where, $\overrightarrow{D'} = \left|\overrightarrow{Y'}(t) - \vec{X}(t)\right|$ indicates the distance of IDS of nth whale from prey's best solution. The shrinking encircling mechanism updates the position of whales during optimization with simultaneous behaviour. The variable p denotes a random number between range of $[0, 1]$. The third phase of search for prey is exploration phase, in addition to randomly adding prey. The humpback whales search randomly depending on each other's position with B greater than -1 or 0. This mechanism allows algorithm to search globally. The mathematical Eqs. (7) and (8) is shown below.

$$\vec{B} = \left|\vec{X}.\overrightarrow{Y_{rand}} - \vec{B}.\vec{E}\right| \tag{7}$$

$$\vec{Y}(t+1) = \overrightarrow{Y_{rand}} - \vec{B}.\vec{E} \tag{8}$$

Where, $\overrightarrow{\mathrm{Y_{rand}}}$ rand is an arbitrarily position vector chosen from current population.

3.3.2. Reptile search algorithm

The RSA involves several steps: initialization of RSA, population, fitness evaluation, encircling phase, and hunting phase. The control parameters should be initialized before executing RSA. The list of control parameters includes V, representing number of attacks, and K being the maximum number of iterations. These parameters aim to achieve balanced abilities during search process. During the phase, the initial solution is generated, as shown in Eq. (9).

$$X_{n,m} = X_m^{min} + rn\ d \times \left(X_m^{max} - X_m^{min}\right), \forall i = 1,2,\ldots\ldots,N,$$
$$\forall m = 1,2,\ldots\ldots,d, \tag{9}$$

Where, $X_{n,m}$ denotes the decision variable of nth solution at mth position. The upper and lower bound decision variables at the mth position are respectively denoted as X_m^{max} and X_m^{min}, which are randomly generated values ranging from 0 to 1. The number of set solutions is denoted by V and stored in X in Eq. (10).

$$X = \begin{bmatrix} X_{1,1}\ X_{1,2}\ \ldots\ldots X_{1,d-1}\ X_{1,d} \\ X_{2,1}\ X_{2,2}\ \ldots\ldots X_{2,d-1}\ X_{2,d} \\ \ldots\ldots \\ X_{V,1}\ X_{V,2}\ \ldots\ldots X_{1,V-1}\ X_{V,d} \end{bmatrix}, \tag{10}$$

Where, each row is represented as is $X_n = \left(X_{n,1}\ X_{n,2}\ \ldots\ldots X_{n,d-1}\ X_{n,d}\right)$, indicating solution at nth position. The fitness values evaluate quality of each solution in population and are evaluated as $f(X_n)\ \forall i = 1,2,\ldots\ldots,V$. The encircling phase involves exploration to extract related information in IDS and finding better solutions by exploring new regions in search space of attack. This includes high walking and belly walking strategies, as described in Eq. (11).

$$X_{n,m}(t+1) = \begin{cases} X_m^{best}(t) - \eta_{n,m}(t) \times \beta - R_{n,m}(t) \times rn\ d, & t \leq \frac{K}{4}, \\ X_m^{best}(t) \times X_{r1,m}(t) \times ES(t) \times rn\ d, & \frac{K}{4} < t \leq \frac{2K}{4}, \end{cases} \tag{11}$$

Where, $X_{n,m}$ denotes the decision variable of nth and mth position, $X_m^{best}(t)$ is mth position in best solution obtained at iteration t. $rn\ d$ is a randomly

generated value ranging between zero and one. $X_{r1,m}(t)$ is decision variable at mth position in r1th solution, where $r1$ denotes range $r1 \in [1, V]$. $\eta_{n,m}(t), P_{n,m}$ and $Avg\ (X_n)$ are evaluated in Eqs. (12) to (14).

$$\eta_{n,m} = X_m^{best} \times P_{n,m} \tag{12}$$

$$P_{n,m} = \alpha + \frac{X_{n,m} - Avg(X_n)}{X_m^{best} \times (X_m^{max} - X_m^{min}) + \epsilon}, \tag{13}$$

$$Avg(X_n) = \frac{1}{d} \sum_{m=1}^{d} X_{n,m}, \tag{14}$$

Where, $P_{n,m}$ denotes the percentage difference between decision variable at mth position of best solution X^{best} and current solution at the same position. The RSA controls exploration ability during hunting, where X_n denotes the current solution and $R_{n,m}(t)$ is a factor that reduces search area in the mth position of nth solution. It assigns randomly decreasing values from 2 to -2 to calculate Eqs. (15) and (16).

$$R_{n,m} = \frac{X_m^{best} - X_{r2,m}}{X_m^{best} + \epsilon} \tag{15}$$

$$ES(t) = 2 \times r3 \times \left(1 - \frac{1}{K}\right) \tag{16}$$

Where, $r2$ represents a randomly generated range of values between 1 and 0, V indicates the index of solution population that is randomly selected, while $r3$ denotes the numerical values between 0 and 1 utilized for transferring relevant features. The hunting phase leverages the exploitation behavior of the RSA, targeting current research regions to seek the optimal solution. It is divided into 2 strategies: hunting cooperation and hunting coordination, as shown in Eq. (17), controlled by $t \leq \frac{3K}{4}$, while hunting cooperation is controlled by the number of attacks.

$$X_{n,m}(t+1) = \begin{cases} X_m^{best}(t) \times rn\ d, & \frac{2K}{4} < t \leq \frac{3K}{4}, \\ X_m^{best}(t) - \eta_{n,m}(t) \times \epsilon - R_{n,m}(t) \times rn\ d, & \frac{2K}{4} < t \leq K. \end{cases} \tag{17}$$

In the RSA, process repeats from the fitness phase to the hunting phase until a high amount of iterations K is reached. The RSA relates to feature selection, providing input to WOA to create the best values.

3.3.3. Proposed whale optimization algorithm with reptile search algorithm

The WOA with RSA method is involved in enhancing intrusion detection by selecting relevant features and optimizing the detection process efficiently to solve complex optimization problems. It is suitable for high-dimensional, nonlinear, and non-convex problems. The goal of the process is to explore the search space of possible solutions, refine attacks, ensure adequate execution time, and achieve an efficient convergence rate. WOA's ability to balance exploration and exploitation to help navigate this large search space effectively. Reptile Search fine-tune the selected features based on their performance, enabling the algorithm to focus on promising regions identified during the exploration phase. The WOA-RSA methods sequentially perform both approaches to achieve a balance between global exploration and local exploitation, leading to more effective optimization.

3.4 Classification

In this research, the current IDS involves DL techniques with characteristics including classification. The proposed method introduces a new detection method for intrusion using MHA and LSTM. In the first phase, the model aims to learn from feature selection in the original vector of IDS [28]. The proposed values are calculated and compared, and a loss function is determined. In the classification phase, the network data is input into the MHA-LSTM to obtain the final prediction result, and calculations are made in the classified phase of the IDS.

3.4.1. Multi-head attention

Following feature selection, the processed data is passed to MHA mechanism model to extract essential features from vector, designed by imitating vision. The index i level from 1 to 40 for the NSLKDD dataset, which has a one-dimensional vector corresponding to each feature. The high network weight is calculated for some features, while others receive low weight. When data enters the MHA model, it is represented as $X = (a_1, a_2, a_3, \dots, a_n), X$, multiplied by the attention weight to obtain values P, Q, and V. The similarity matrix, different from the feature selection obtained, is derived by multiplying P and Q^X. After similarity is normalized using the SoftMax function, calculations in IDS are reduced to a certain extent. V is then multiplied to obtain network information of the same dimension as input

data, as shown in the Eqs. (18) to (20), which determines the dimension of the matrix values.

$$head_m(P,Q,V) = softmax\left(\frac{PQ^X}{\sqrt{d_k}}\right)V \quad (18)$$

$$softmax(Y_n) = \frac{exp(Y_n)}{\sum_m exp(Y_n)} \quad (19)$$

$$T = Concat\left(head_1, head_2, head_3\right) \quad (20)$$

Where, Y_n denotes dimension of k matrix. The MHA is embedded and then incorporated into the network with dropout to enhance generalization ability of proposed method. By utilizing MHA mechanism alongside LSTM, it captures vector dependencies over longer distances, thereby enhancing accuracy and efficiency in identifying network intrusions.

3.4.2. Long short-term memory

After data is weighted by attention mechanism and provided to LSTM model. The LSTM, a type of RNN learns and remembers long-term dependencies, capturing relationships between different features in vector. The gradient problem disappears, thus improving the classification accuracy using LSTM. The two-dimensional vector composed in the MHA is denoted by T and involves $T = \left(m_1, m_2, \ldots, m_3\right)$. There are three gates used in this structure: the hidden layer, input, and output. Initially, $h(t-1)$ and $c(t)$ are initialized at time t. The mathematical formula is shown in Eqs. (21) to (24) below.

$$f(t) = \sigma\left(D_f h_{t-1} + H_f m_t + b_f\right) \quad (21)$$

$$n(t) = \sigma(D_n h_{t-1} + H_n m_t + b_n) \quad (22)$$

$$a(t) = \tanh(D_a h_{t-1} + H_a m_t + b_a) \quad (23)$$

$$p(t) = \sigma\left(D_p h_{t-1} + H_p m_t + b_p\right) \quad (24)$$

Where, h_{t-1} denotes hidden layer status values at time ($t-1$), and D_f, D_n, D_a are denoted as weight parameters of h_{t-1} in feature selection process of forget gate. The related function is shown in Eqs. (25) and (26).

$$\tanh(x) = \frac{1-e^{-2x}}{1+e^{-2x}} \quad (25)$$

$$h(t) = p(t) \odot \tanh\left(c(t)\right) \quad (26)$$

The result of forget gate and output gate is evaluated within the range of $c(t-1)$ which constitutes cell state $c(t)$ at the moment of the Eq. (26). The final hidden state equation is denoted by $\odot$ representing the Hadamard product.

3.4.3. Multi head attention with long short term memory

The MHA with LSTM is combined to perform IDS. Independent attention outputs are then concatenated and linearly transformed to obtain the final output. Given an input sequence, the MHA computes attention scores for each position in the sequencely performed in the IDS to transfer information efficiently. The LSTM, designed to handle sequential data, contains memory cells capable of storing information across long sequences. They use input, forget, and output gates to control the flow of data within the network. The combination of these methods allows for fine-grained analysis of log entries, focusing on attack patterns indicative of malicious activity. The incorporation of MHA-LSTM equips the model to adeptly manage extensive datasets and complex tasks while upholding superior performance and accuracy. This combined methodology involves a robust strategy to detect and classify attacks effectively, particularly suited for large-scale network intrusion detection.

4. Experimental result

This section ensures the accuracy of experiment analysis using datasets namely, CICIDS2018 and NSL-KDD to carry out a more comprehensive evaluation of the proposed WOA-RSA-based MHA-LSTM method. The implementation of the proposed method is carried out using Python 3.10.12, Windows 10 (64-bit) - Operating System, an Intel Core i5 processor, and RAM-8GB. The performance measures used for evaluation and the results of the feature selection and classification are explained in section 4.1. The performance of the proposed method is evaluated using different performance metrics including Accuracy, Precision, F1-score and Recall defined by Eqs. (27) to (30).

$$Accuracy = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (27)$$

$$Precision = \frac{TP}{(TP+FP)} \quad (28)$$

$$F1-score = 2 * \frac{(Precision*Recall)}{Precision+Recall} \quad (29)$$

$$Recall = \frac{TP}{TP+Fn} \quad (30)$$

Where, $TP, TN, FP,$ and FN signify the True Positive, True Negative, False Positive, and False Negative values, respectively. The proposed method, involving feature selection using WOA-RSA and classification using MHA-LSTM performs with high accuracy to detect the attack.

4.1 Performance analysis

In this section, the proposed method involving feature selection and classification processes is evaluated using different performance metrics, including Accuracy, Precision, F1-score, and Recall for the CICIDS2018 and NSL-KDD datasets. The feature selection process with datasets is represented in Tables 1 and 2, which describe feature selection results. The classification process with datasets is represented in Tables 3 and 4, which display the classification results. Figs. 2 and 3 illustrate the results in terms of accuracy and loss graphs, while Figs. 4 and 5 depict the results of the confusion matrix where ROC is a crucial parameter in IDS.

The performance of WOA-RSA feature selection is evaluated based on accuracy, precision, F1-score, and recall on the CICIDS2018 dataset, as described in Table 1. The existing methods using feature selection techniques such as Grey Wolf Optimizer (GWO), RSA and WOA are also evaluated. The WOA-RSA method achieves a high accuracy precision, recall and F1-score of 99.99%. The feature selection technique RSA with WOA attains high accuracy, reaching 99.99% because it selects the related features that enables to easily detect attacks.

The performance of WOA-RSA feature selection is evaluated based on accuracy, precision, F1-score, and recall on the CICIDS2018 dataset, as described in Table 2. The existing methods using feature selection techniques such as GWO, RSA, and WOA are also evaluated. The WOA-RSA method achieves a high accuracy, precision, recall and F1-score of 99.99%. The feature selection technique WOA with RSA achieves a high accuracy of 99.99%, because it extracts related features that easily detect attacks.

The performance of MHA-LSTM classification is evaluated based on accuracy, precision, F1-score, and recall on the CICIDS2018 dataset, as described in Table 3. The existing methods using feature selection techniques such as RNN, DNN, and LSTM are also evaluated. The MHA-LSTM method attains a high accuracy, precision, recall, and F1-score, all at 99.99%. The classification technique is MHA with LSTM, achieving high accuracy, reaching 99.99% because of its robustness and efficiency.

4.1.1. Epoch v/s accuracy and epoch v/s loss using CICIDS 2018 dataset

Figs. 2 and 3 illustrate the results in terms of accuracy and loss graphs. These graphs indicate that the accuracy of the proposed model improves over time. When initial accuracy is low, it consistently increases with each epoch. This trend is attributed to the learning capabilities of the proposed model, which gradually stabilizes and becomes more adept at distinguishing between normal and malicious data.

4.1.2. Confusion matrix and ROC curve analysis using CICIDS 2018 dataset

Figs. 4 and 5 display the results of the confusion matrix, and ROC is a crucial parameter in IDS. The confusion matrix analyses classification and measures the accuracy of the proposed method in distinguishing between benign and attack instances. The ROC curve visually represents performance by comparing the false positive rate and the true positive

Table 1. Performance of the feature selection process on the CICIDS2018 dataset

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
GWO	89.55	90.00	88.00	90.00
RSA	95.67	96.00	94.00	95.00
WOA	92.33	92.78	92.30	92.50
DOA	94.25	94.35	94.13	94.85
Proposed Feature selection (WOA-RSA)	99.99	99.99	99.99	99.99

Table 2. Performance of feature selection process on NSL-KDD dataset

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
GWO	89.00	90.00	88.00	89.00
RSA	95.677	96.00	94.00	95.00
WOA	91.23	92.58	91.00	91.30
DOA	94.25	94.35	94.13	94.85
Proposed Feature selection (WOA-RSA)	99.99	99.99	99.99	99.99

Table 3. Performance of classification process on the CICIDS2018 dataset

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
RNN	96.59	97.00	97.00	96.00
DNN	96.44	96.00	96.00	96.00
LSTM	94.95	95.00	95.00	95.00
Proposed Method (MHA-LSTM)	99.99	99.99	99.99	99.99

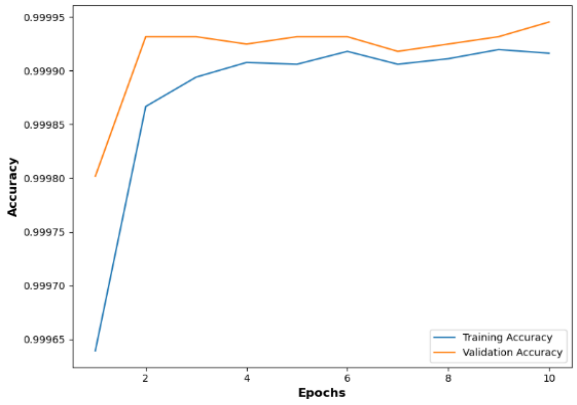

Figure. 2 Result of the accuracy

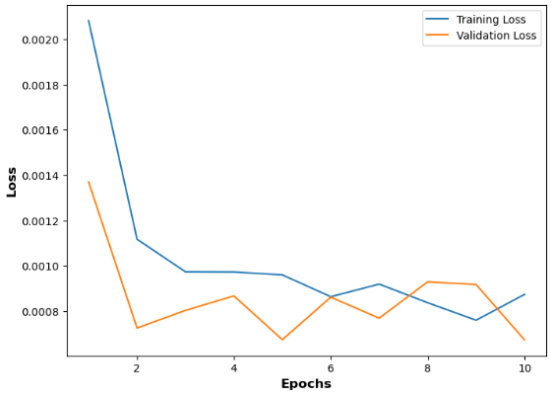

Figure. 3 Result of the loss graph

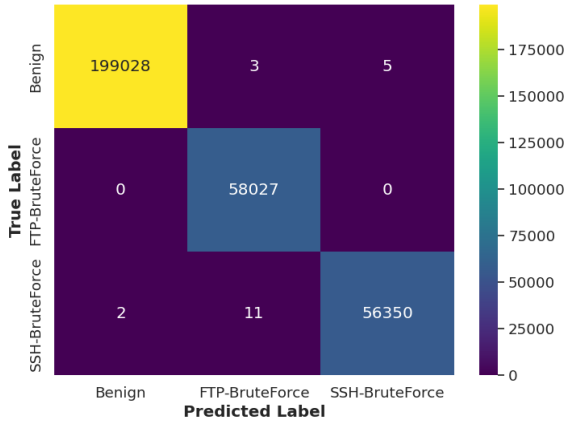

Figure. 4 Result of the confusion matrix

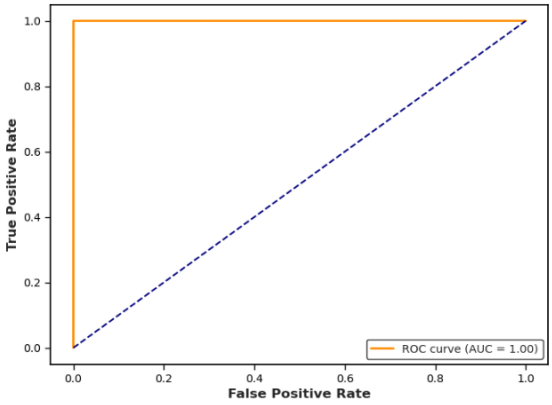

Figure. 5 Result of the ROC curve

Table 4. Performance of classification process on NSL-KDD dataset

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
RNN	98.55	98.47	98.55	98.46
DNN	97.50	97.39	97.22	97.22
LSTM	98.69	98.69	98.69	98.69
Proposed Method (MHA-LSTM)	99.99	99.99	99.99	99.99

rate. The proposed method significantly outperforms other existing methods.

The performance of MHA-LSTM classification is evaluated in terms of accuracy, precision, F1-score, and recall on NSL KDD dataset, as described in Table 4. The existing methods using feature selection techniques such as RNN, DNN, and LSTM are also evaluated. The MHA-LSTM method accomplishes a superior accuracy, precision, recall, and F1-score, all at 99.99%. The classification technique is MHA with LSTM attains a commendable accuracy of reaching 99.99%, rendering robustness and efficiency.

4.1.3. Epoch v/s accuracy and epoch v/s loss using NSL-KDD dataset

Figs. 6 and 7 illustrate results in terms of accuracy and loss graphs. These graphs indicate that accuracy of proposed model improves over time. When initial accuracy is low, it consistently increases with each epoch. This trend is attributed to the learning capabilities of the proposed model, which gradually stabilizes and becomes more adept at distinguishing between normal and malicious data.

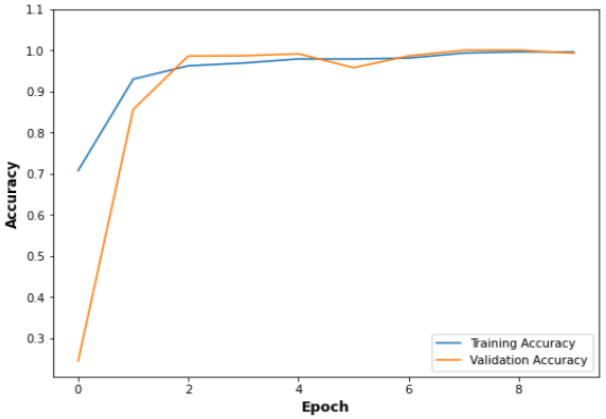

Figure. 6 Result of the accuracy

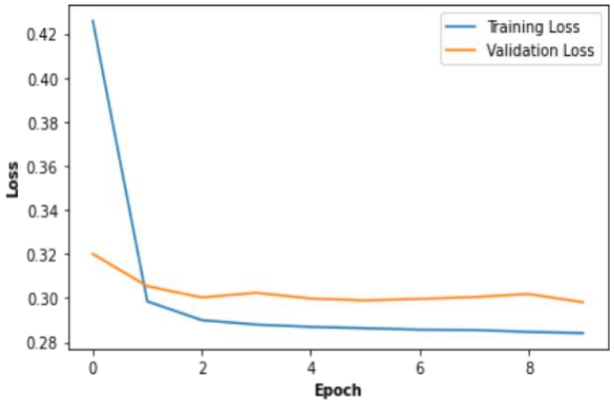

Figure. 7 Result of the loss graph

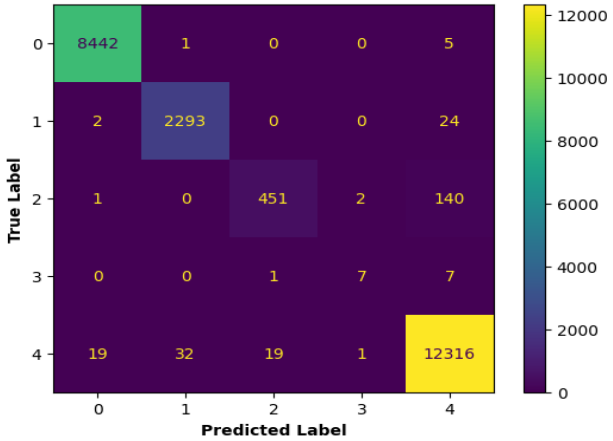

Figure. 8 Result of the confusion matrix

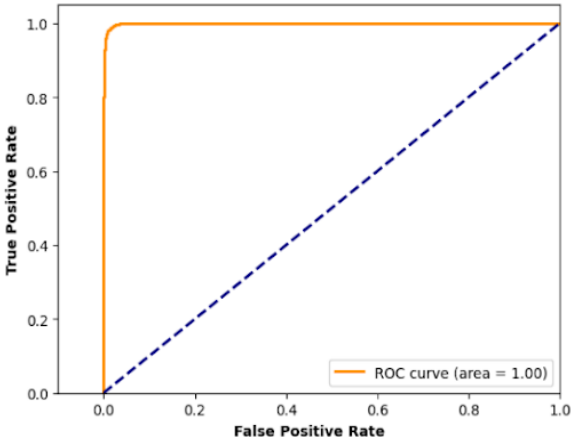

Figure. 9 Result of the ROC curve

4.1.4. Confusion matrix and ROC curve analysis using NSL-KDD dataset

Figs. 8 and 9 display the results of the confusion matrix, and ROC is a crucial parameter in IDS. The confusion matrix analyses classification and measures the accuracy of the proposed method in distinguishing between benign and attack instances. The ROC curve visually represents the performance by comparing the false positive rate and true positive rates. The proposed method significantly outperforms other existing methods.

4.2 Comparative analysis

The performance of proposed method WOA-RSA is compared with existing methods, including RSA [16], LeNet-based IDS [17], DNN [18], and LSTM [19]. The comparative analysis involves two datasets: CICIDS2017, CICDS2018 and NSL-KDD. In this research, the proposed WOA-RSA method achieves high accuracy, accomplishing accuracies of 99.99% on the CICIDS2017 of 99.997%, CICIDS2018 dataset and 99.99% on the NSL-KDD dataset, respectively. Table 5 describes a comparative analysis of proposed method.

4.3 Discussion

In this section, limitations and advantages of proposed method are discussed and compared with that of the existing methods. For example, RSA [17] method requires increased computational resources for packet analysis during attack detection. LeNet-based IDS [18] is affected by the complexity of the training data, leading to inaccuracies in classifying attacks. DNN [19] consumes maximum time to train the model and enhances the computational cost. LSTM-AE [20] struggles when the input data is not sequential, slowing down the training process and hindering attack detection. The DT [22] technique struggle to identify intrusions effectively because they consider the various attack leading to high false-negative rates. In this research, RSA –WOA - MHALSTM is combined to extract relevant features and find the best values. This process explores the search space of possible solutions, refining attacks with adequate execution time and an efficient

Table 5. Comparative analysis of the proposed method

Datasets	Method	Accuracy (%)	Precision (%)	F1-Score (%)	Recall (%)
CICIDS2017	RSA [16]	99.996	99.996	99.996	99.996
	LSTM –AE [19]	99.99	99.99	99.99	99.99
	DT [21]	94.72	98.38	86.93	99.33
	Proposed WOA-RSA- MHALSTM method	**99.997**	**99.997**	**99.997**	**99.997**
CICIDS2018	DNN[18]	99.21	99.21	99.21	99.20
	LSTM –AE [19]	99.10	99.07	99.02	99.10
	Proposed WOA-RSA- MHALSTM method	**99.99**	**99.99**	**99.99**	**99.99**
NSL-KDD	RSA [16]	99.23	99.23	99.92	99.92
	LeNet based IDS [17]	99.28	99.21	N/A	N/A
	Proposed WOA-RSA–MHALSTM method	**99.99**	**99.99**	**99.99**	**99.99**

convergence rate. The MHA-LSTM using classification enables fine-grained analysis of log entries, focusing on the attack patterns indicative of the malicious activity.

5. Conclusion

In this research, the WOA-RSA algorithm is proposed for feature selection and MHALSTM for classification to improve the performance of IDS while minimizing the number of features needed to build a robust IDS. Separating the features allows for accurate attack detection and extraction of relevant features to select the best values and reduce data dimensionality. The WOA-RSA methods sequentially perform both approaches to accomplish a balance between local exploitation and global exploration, leading to more effective optimization. This ensures an effective scaling of the MHA-LSTM model's classification capability across CICIDS2018 and NSL-KDD datasets, handling the maximum complex tasks without compromising on the performance and accuracy. This proposed approach achieves a commendable accuracy of 99.997% on CICIDS2017, 99.99% on CICIDS2018 and 99.99% on NSL-KDD datasets, as opposed to DNN and LSTM. Future research can effectively utilize anomaly detection in deep learning to minimize time consumption and improve the accuracy in detecting attacks.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

The paper conceptualization, methodology, software, validation, formal analysis, investigation, resources, data curation, writing—original draft preparation, writing—review and editing, visualization, have been done by 1st author. The supervision and project administration, have been done by 2nd author.

Notation

Notation	Description
s	standard deviation
μ	Mean
$\vec{B}$ and $\vec{E}$	coefficient vectors
X	Position Vector
$\vec{B}$	Spiral updating position and decreasing values
$\overrightarrow{D'} = \left\|\overrightarrow{Y'}(t) - \vec{X}(t)\right\|$	IDS of nth whale
$\overrightarrow{Y_{rand}}$	Random position vector
V	Control parameter
K	Maximum number of iterations
X_m^{max} and X_m^{min}	Upper and lower bound decision variable at mth position
$X_{n,m}$	Decision variable
$X_m^{best}(t)$	Position of best solution
$P_{n,m}$	Percentage
X_n	Current solution
$r2$, r3	Randomly generated range
$X = (a_1, a_2, a_3, \ldots, a_n)$,	Multiplied values
P, Q, and V	Attention weight
Y_n	Dimension
T	Two-dimensional vector
$h(t-1)$ and $c(t)$	Initialized at time
D_f, D_n, D_a	Weight
$c(t-1)$	Forget and output gate
TP, TN, FP, and FN	rue Positive, True Negative, False Positive, and False Negative values,

References

[1] S. Ullah, J. Ahmad, M.A. Khan, E.H. Alkhammash, M. Hadjouni, Y.Y. Ghadi, F. Saeed, and N. Pitropakis, "A New Intrusion Detection System for the Internet of Things via Deep Convolutional Neural Network and Feature Engineering", *Sensors*, Vol. 22, No. 10, p. 3607, 2022.

[2] X.H. Nguyen, X.-D. Nguyen, H.-H. Huynh, and K.-H. Le, "Realguard: A Lightweight Network Intrusion Detection System for IoT Gateways", *Sensors*, Vol. 22, No. 2, p. 432, 2022.

[3] W.W. Lo, S. Layeghy, M. Sarhan, M. Gallagher, and M. Portmann, M., "E-graphsage: A graph neural network-based intrusion detection system for IoT", In: *Proc. of NOMS 2022-2022 IEEE/IFIP Network Operations and Management Symposium*, Budapest, Hungary, pp. 1-9, 2022.

[4] A. Awajan, "A Novel Deep Learning-Based Intrusion Detection System for IoT Networks", *Computers*, Vol. 12, No. 2, p. 34, 2023.

[5] A. Basati, and M.M. Faghih, "PDAE: Efficient network intrusion detection in IoT using parallel deep auto-encoders", *Information Sciences*, Vol. 598, pp. 57-74, 2022.

[6] O.A. Alghanam, W. Almobaideen, M. Saadeh, and O. Adwan, "An improved PIO feature selection algorithm for IoT network intrusion detection system based on ensemble learning", *Expert Systems with Applications*, Vol. 213, Part A, p. 118745, 2023.

[7] M. Sarhan, S. Layeghy, and M. Portmann, "Towards a standard feature set for network intrusion detection system datasets", *Mobile Networks and Applications*, Vol. 27, No. 1, pp. 357-370, 2022.

[8] 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*, Vol. 10, No. 1, pp. 205-216, 2024.

[9] A.R. Khan, M. Kashif, R.H. Jhaveri, R. Raut, T. Saba, and S.A. Bahaj, "Deep learning for intrusion detection and security of Internet of things (IoT): current analysis, challenges, and possible solutions", *Security and communication networks*, Vol. 2022, No. 1, p. 4016073, 2022.

[10] M. Mohy-Eddine, A. Guezzaz, S. Benkirane, and M. Azrour, "An efficient network intrusion detection model for IoT security using K-NN classifier and feature selection", *Multimedia Tools and Applications*, Vol. 82, No. 15, pp. 23615-23633, 2023.

[11] N. Yadav, S. Pande, A. Khamparia, and D. Gupta, "Intrusion detection system on IoT with 5G network using deep learning", *Wireless Communications and Mobile Computing*, Vol. 2022, No. 1, p. 9304689, 2022.

[12] Y. Li, S. Ghoreishi, and A. Issakhov, "Improving the accuracy of network intrusion detection system in medical IoT systems through butterfly optimization algorithm", *Wireless Personal Communications*, Vol. 126, No. 3, pp. 1999-2017, 2022.

[13] V. Ravi, R. Chaganti, and M. Alazab, "Deep learning feature fusion approach for an intrusion detection system in SDN-based IoT networks", *IEEE Internet of Things Magazine*, Vol. 5, No. 2, pp. 24-29, 2022.

[14] M.S.A. Muthanna, R. Alkanhel, A. Muthanna, A. Rafiq, and W.A.M. Abdullah, "Towards SDN-enabled, intelligent intrusion detection system for internet of things (IoT)", *IEEE Access*, Vol. 10, pp. 22756-22768, 2022.

[15] A. Kumar, K. Abhishek, M.R. Ghalib, A. Shankar, and X. Cheng, "Intrusion detection and prevention system for an IoT environment", *Digital Communications and Networks*, Vol. 8, No. 4, pp. 540-551, 2022.

[16] A. Dahou, M. Abd Elaziz, S.A. Chelloug, M.A. Awadallah, M.A. Al-Betar, M.A.A. Al-Qaness, and A. Forestiero, "Intrusion detection system for IoT based on deep learning and modified reptile search algorithm", *Computational Intelligence and Neuroscience*, Vol. 2022, No. 1, p. 6473507, 2022.

[17] D. Selvapandian, and R. Santhosh, "Deep learning approach for intrusion detection in IoT-multi cloud environment", *Automated Software Engineering*, Vol. 28, p. 19, 2021.

[18] 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, 2022.

[19] 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, 2023.

[20] M. Keshk, N. Koroniotis, N. Pham, N. Moustafa, B. Turnbull, and A.Y. Zomaya, "An explainable deep learning-enabled intrusion detection framework in IoT networks", *Information Sciences*, Vol. 639, p. 119000, 2023.

[21] A.S. Jaradat, M.M. Barhoush, and R.B. Easa, "Network intrusion detection system: Machine

learning approach", *Indonesian Journal of Electrical Engineering and Computer Science*, Vol. 25, No. 2, pp.1151-1158, 2022.

[22] K. Kaabneh, I. AbuFalahah, K. Eguchi, S. Gochhait, I. Leonova, Z. Montazeri, and M. Dehghani, "Dollmaker Optimization Algorithm: A Novel Human-Inspired Optimizer for Solving Optimization Problems", *International Journal of Intelligent Engineering & Systems*, Vol. 17, No. 3, pp. 816-828, 2024.

[23] CICIDS2017 dataset: https://www.kaggle.com/datasets/kk0105/cicids2017 (Accessed on April 2024).

[24] CICIDS2018 dataset: https://www.kaggle.com/datasets/jvageesh11/cicids2018 (Accessed on April 2024).

[25] NSL-KDD dataset: https://www.kaggle.com/datasets/dhoogla/nslkdd (Accessed on April 2024).

[26] M.A. Siddiqi, and W. Pak, "An agile approach to identify single and hybrid normalization for enhancing machine learning-based network intrusion detection", *IEEE Access*, Vol. 9, pp. 137494-137513, 2021.

[27] T.A. Siahmarzkooh, and M. Alimardani, "A Novel Anomaly-based Intrusion Detection System using Whale Optimization Algorithm WOA-Based Intrusion Detection System", *International Journal of Web Research*, Vol. 4, No. 2, pp. 8-15, 2021.

[28] J. Zhang, X. Zhang, Z. Liu, F. Fu, Y. Jiao, and F. Xu, "A Network Intrusion Detection Model Based on BiLSTM with Multi-Head Attention Mechanism", *Electronics*, Vol. 12, p. 4170, 2023.