Diyala Journal of Engineering Sciences

Journal homepage: https://djes.info/index.php/djes

ISSN: 1999-8716 (Print); 2616-6909 (Online)

Enhancing Intrusion Detection Systems Using Metaheuristic Algorithms

Heba Mohammed Fadhil[1*], Zinah Osamah Dawood[1] and Ammar Al Mhdawi[2]

[1]Department of Information and Communication, Al-Khwarizmi College of Engineering, University of Baghdad, Baghdad, Iraq.
[2]School of Engineering and Sustainable Development, De Montfort University Leicester, UK.

ARTICLE INFO

Article history:
Received April 20, 2024
Revised July 19, 2024
Accepted July 27, 2024
Available online September 1, 2024

Keywords:
Intrusion detection system
Metaheuristic algorithms
Lion optimization algorithm
Grey wolf optimization
Hyperparameter
Feature selection
Deep learning

ABSTRACT

In the current network security framework, Intrusion Detection Systems (IDSs) happen to be among the major players in ensuring that the network activity is being monitored round the clock for any intrusions which may occur. The rising degree of cyber threats' intricacy enforces the constant development of IDS methodologies to maintain effectiveness in detecting and reversing the emergence of any extra risks. Therefore, to settle the matter featured by, this research studies try to incorporate the most powerful metaheuristic algorithms, Lion Optimization Algorithm (LOA) and Grey Wolf Optimizer (GWO) in particular, to develop better detection accuracy and efficiency. The core obstacle recognized in this article is the fact that many systems of IDS send out false alarms and their mechanisms of detection of the true anomalies need to be improved immensely. In a nutshell, the change would unveil a fresh way of using LOA and GWO using them to promote the enhancement of internet defences systems in real-time. These schemes can discover previously unknown weaknesses or stealthy attacks. The core of this undertaking would consist in the conception and implementing of a Hybrid Network Intrusion Detection System, which will be created by blending the Lion Optimization Feature Selection (LOFS) and GWO smelters, denoted as LOFSGWO. Critically, the main purpose is to incorporate the GWO as a tool in the operations to cut down the dangerous parameters favourable towards an intrusion mechanism in the framework of a Hybrid CNN-LSTM Deep Learning system. Model tests reveal over 99.26% accuracy of low negative samples into out of a box that are served as testing as well as NSL-KDD dataset, which are similar to the simulation of WUSTL-EOM 2020 system. The obtained outcomes verify the relevance and efficiency of the suggested strategy, which may be used in the resolution of the issues faced in a network security today.

1. Introduction

Amid the linkage between the digital and real worlds, information security remains a major issue, especially when it appears globally. Despite this passing time, the tremendous growth in cybersecurity attacks remains the leading factor in the introduction of cutting-edge solutions to cyber defense [1]. The network intrusion detection system (NIDS) determines its role as a major reserve in the shield of aids; it marches along with the speed of breaches and stops intrusions before they occur [2].

Even though there are steadily evolving cybersecurity technologies, the persistent threat of cyberattacks such as Denial of Service (DoS) is something to be mindful of. This also applies to computer viruses and data breaches, which are enormous threats to computer networks [3]. In fact, IDSs have moved from valuable items to

* Corresponding author.
E-mail address: heba@kecbu.uobaghdad.edu.iq
DOI: 10.24237/djes.2024.17302

organizations meant to eliminate risks. These systems have great help in different detection forms, including signature-based, misuse-based, and anomaly-based NIDS, each of which is used to discover a certain type of threat [4].

The detection of anomalies can probably contribute to discovering new forms of attacks by pinpointing the obvious changes from predefined routine actions. Such an ability becomes more crucial when compared to behavior-based detection systems that aim to mitigate the growing threats in concern with, not using based detection [5]. Although this technique may perform differently depending on whether the underlying dataset is complex or dimensional, the user must find an optimized approach to overcome these challenges, thus ensuring the successful completion of such tasks.

In recent years, the combination of Machine Learning (ML) and Deep Learning (DL) technologies has been highly beneficial in this area, although it has also raised ethical concerns [6]. ML techniques, in particular, enable computers to go beyond just programming to identify attacks in network traffic in cases where incidents and exploits may otherwise have gone unnoticed [7]. ML algorithms can face ineffectiveness in the case of noisy or irrelevant features among the dimensions, which indicates that the Feature Selections (FS) approaches and methods should be used in data Feature Selections [8]. Through FS, the DL system builds constructive and interpretable models by determining the most meaningful and relevant features that reduce the inaccuracy rate[9]. Simultaneously, DL approaches have a unique advantage in learning the complex features of traffic flow and temporal correlations from such data [10]. The combined approach of FS with DL can offer the detection mission a higher margin of reliability, and hence, give this mission greater velocity and accuracy in detecting cyber-crime [11].

This study developed the LOFSGWO model, a new system for intrusion detection based on the need for effectively robust solutions instead of the typical existing methods. Utilization of the Lion Optimization Feature Selection (LOFS) approach as well as a Grey Wolf Optimizer (GWO) for hyperparameter optimization and improvement of the algorithm performance as much as possible. Hence, this technique seeks to reduce false alarms in addition to network traffic upsurges. The key contributions of this study are as follows:

1. This investigation of a NIDS approach is based on lion optimization feature selection for deep learning (LOFSGWO). This is a combination of the classifier and LOFS, and also uses GWO technology for hyperparameter refinement. Feature selection along with a hyperparameter-tuned deep learning model will contribute to an increase in the accuracy and overall capability of intrusion detection systems.
2. The feature selection technique LOFS is implemented to pinpoint and extract the pertinent features (i.e., those that are the most relevant and informative) from the dataset. The involvement of LOFS in the study was made so that NIDS models are not just up-to-date, but also easy to understand by using the most distinguishable features to shorten the process of detection.
3. The GWO algorithm was used for additional tuning. The goal here is to find the best configuration of hyperparameters that would ideally result in higher detection accuracy and greater effectiveness of the network protection mechanism in spotting intrusions.

2. Literature review

This research details a new method of Intrusion Detection Systems (IDS) designed for IoT devices, through which DL methods are utilized. Furthermore, an FS-DNN approach was postulated as a solution that focuses on lifting the burden of the collision of highly interconnected features. It focuses on parameter tuning and hyperparameter tuning for performance optimization [12]. Mohy-eddine et al. [13] designed an effective NIDS applicable to IoT platforms involving both the feature selection procedure and KNN method. Different FS techniques, including the GA, academic and google scholar univariate statistical tests, and PCA, will be used to enhance the performance of Intrusion Detection Systems. out of all the enumerated results of the considered method, Featured, which is associated with the ten most effective factors, stands out.

A previous study proposed a hybrid approach to detecting intrusion in IoT devices based on both Deep Learning (DL) and shallow learning techniques, which reviewed the applicability of a synergistic that integrates DL and shallow learning techniques to detect intrusion on the IoT devices. With the help of this algorithm, the SMOFS Feature Selection strategy is applied for selection of a set of parameters larger compared to the specified group, but containing only closely related parameters regarding to the aspect discussed during estimation of the first set. In addition, Siamese Neural Network configuration was intended to improve the data distinguishable geometric transforms deployed in order to optimize the increase in intrusion detection systems' performance [14]. For instance, among the many works done by the researchers Syed et al. [15] presented a new Generation IDS situated in the fog-cloud for Internet of Things (IoTs) environment. The SW includes a distributed processing architectural design; the primary goal is to filter data according to the type of attackers. However, the existing studies do not include an FS stage for transforming the IoT data into smarter and more accurate while also minimizing intrusiveness. Recurrent Neural Networks (RNNs) are deep learning techniques that include RNNs, such as simple RNN and progressive Bidirectional Long Short-Term Memory (BiLSTM). These are used to identify attacks and are helpful in creating intrusion-detection systems.

In this extensive review, numerous IDS and AI methods for securing IoMT are examined. This paper classifies IDS schemes, introduces datasets that are used for analyses, cybersecurity threats, and presents cloud-fog-edge architectures. The legal and moral issues of IoMT security are also discussed, making the picture more complete. The paper also noted that advanced intrusion detection systems and new AI approaches are necessary to tackle evolving cybersecurity challenges in IoMT, with due attention paid to both legal aspects and ethical questions [16]. Chen et al. [17] deals with the problem of malicious manipulation of healthcare data in Healthcare IoMT Systems. A federated data sanitization defense method is proposed in combination with both the concepts of TFL and clustering to filter poisoned addition. The proposed approach shows the strength to different data poisoning attacks and hence conserves medical-assisted diagnosis models robustness.

With the introduction of the Internet of Medical Things (IoMT) in healthcare, there have been new advances in security and privacy issues. Research studies and contrasts different ML algorithms for intrusion detection in IoMT networks. For the Bot-IoT benchmark datasets, we analyzed ML algorithms, including k-nearest neighbor (KNN), Naïve Bayes (NB), Support Vector Machine (SVM), Artificial Neural Network (ANN), and Decision Tree (DT). These results demonstrate the effectiveness of ML algorithms in detecting and blocking intruders in IoMT networks [18]. As ICT continues to evolve, IoMT has emerged. However, security issues occur because of possible cyber-attacks. This paper discusses a DRNN approach and supervised machine-learning models that are efficient for cyber-attack detection in the IoMT setting. The proposed models, such as Random Forest (RF), DT, KNN, and ridge classifier, have better performance with an accuracy of 99.76% [19].

The security of IoMT is a major issue, and this study aimed to detect malicious traffic in

IoMT settings. This study suggests the use of the XGBoost classification algorithm for accurately classifying malicious traffic. This work also compares machine learning algorithms such as Decision Tree, Random Forest, Logistic Regression (LR), Supporting Vector Machine (SVM), Naïve Bayes (NB), and Multilayer Perceptron (MLP) and it is evident that XGBoost has a 100 % accuracy [20].

Jithish et al. [21] proposed fordetecting intrusions in cyber-physical manufacturing systems (CPMS) using Kernel Principal Component Analysis (KPCA) and Self-Organizing Maps. This new approach interprets complicated information into a hyperplane feature space that improves the measure of specificity on pattern identification as well as the identification of invasions. This paper also uses the continuous stirred tank reactor (CSTR) model for evaluating the effectiveness of this method through simulations. The results shown implicated the accuracy rate of the proposed technique to be at100% being closer to the actual mark of 95%. 5 % more compared to alter-insertion based intrusion detection methods, which are more frequently used in nowadays network environments. Similarly, in the study by Maseno and Wang [22], an improved ELM-SVM model with a sequential feature selection method was presented. The problem encountered with ELM is that it is extremely sensitive to the choice of the input parameters that has influence with its performance. To this end, to further enhance the ELM's overall performance, the weights of ELM were again improved by using a genetic algorithm (GA).

Once optimization is done then this algorithm works as an estimator to select the best set of features with the help of other methods of feature selection algorithms vs sequential forward selection also known as the-wrapper technique. Out of all these features, the selected ones were interpreted and applied for classification purposes by using SVM. To assess the performance of the proposed model, IoT-ToN and UNSWNB15 datasets were applied to check the model performance. Analyzing the results of the experiments presented it has been observed that there is an excellent identification of intrusion for the proposed model using the IoT-ToN network dataset with an accuracy of 99% and UNSWNB15 dataset with an accuracy of 86%. In light of the above research outcomes, it can be concluded that the proposed model is a viable tool for boosting the capability of the IDS datasets.

Table 1: Contrasting similar studies

Study	Methodology	Limitations
[12]	DL methods with FS-DNN	Lack of discussion needs explanations of hyperparameter tuning No Comparison with other methods
[13]	The NIDS models that apply feature selection using genetic algorithms (GA), and principal component analysis (PCA)	Limited explanation of feature selection methods Performance is not comparable to other techniques
[14]	Hybrid approach that comprises (deep-learning and shallow learning) and SMOFS tree	The case where different shallow learning methods are selected from has limited explanation
[15]	Generation of Fog IDS for IoT systems in fog-cloud	There is no discussion of the success of the FS stage or the interoperability of the complete solution.
[16]	Different IDS techniques and AI will pertain to security IoMT	Inadequate selection of datasets would be the hindrances that will be demonstrated Algorithms Evaluation metrics are hardly discussed
[17]	Federated data which refers to sanitization of healthcare IoMT	Insufficiency in the effect's explanation of TFL and clustering. It surpasses the speed or complexity of human data sanitization.

[18]	ML algorithms for intrusion detection in IoMT networks	It surpasses the speed or complexity of human data sanitization. No Comparison with other methods
[19]	DRNN and supervised ML models for detection of cyber-attack	Neither ensemble methods nor the techniques based on single voters
[20]	XGBoost for malicious traffic detection in IoMT	The observation only comparison with various classifiers was not done.
[21]	KPCA and SOMfor detecting intrusions in cyber-physical	The feature selection methods are not adequately explained
[22]	Optimized extreme learning machine (ELM) with a support vector machine (SVM) classifier, along with a sequential feature selection technique	One of the primary difficulties associated with ELM lies in choosing appropriate input parameters.

The summary also reveals the approaches mentioned in Table 1 which addresses the development of IDS for IoT devices using DL and the proposed FS-DNN to address interconnections of features. To enhance accuracy in IDS, other approaches as the GA, ANOVA, and PCA, have been applied. Further, this review also considers the application of the DL together with shallow learning like Siamese Neural Network and fog cloud architecture for improvement of performance. AI solutions to safeguard IoMT security have been discussed in detail based on the following approaches: Federated Learning, clustering and Machine learning algorithms like KNN, SVM and XGBoost yielded significant accuracy ratios. But there are several drawbacks at present, including coping with high false positive rates, choosing parameters that are perfect, finding models that are stable and versatile enough for new threats.

However, there are restrictions that apply even with more modern IDS and AI methods, and these are as follows. Among these are the issues of handling a higher probability of false positives, the difficulty in fine-tuning system parameters and the need to employ robust and adaptive models to counter new-age cyber threats. Also, the absence of a general methodological approach might not allow the application of the proposed methods across a scale of various datasets and different scenarios.

3. Proposed Approach

The Proposed Lion Optimization Feature Selection Grey Wolf Optimizer (LOFSGWO) model fuses with LOA and GWO to improve IDS efficiency in finding network anomalies. First, the traffic dataflow is preprocessed for the normalization of values and missing data filling, and then Lifted Orthogonal Feature Selection is used to pick up the relevant features, resulting in the streamlining of the process, while simultaneously improving the efficiency and accuracy of the IDS. Starting positions and velocities of grey wolves and lions along with predetermined values of global and local optima that activate the initial operation and feature selection procedures. The model gradually attains weights for positions and velocities by running iterations and fitness values calculated from the network traffic (as a model) until convergence. The parameters obtained at the end of the learning optimization are used to feed into a CNN-LSTM model that has the ability to learn the complex patterns and long-term temporal dependencies from the data. The model was evaluated using the test data to determine its accuracy, precision, recall, and F1. This indicates whether the model can identify whether the network has a problem, as shown in Algorithms (1) and Figure 1.

Algorithm 1 LOFSGWO Algorithm

Input: Dataset
Output: Model Performance
1: Preprocess data (normalization, handling missing values)
2: Apply Lion Optimization Feature Selection (LOFS) to select relevant features
3: Initialize Grey Wolf Optimization (GWO) parameters (e.g., population size, maximum iterations)
4: Initialize Lion Optimization Algorithm (LOA) parameters (e.g., pride size, exploration factor)
5: Initialize grey wolf pack positions and velocities
6: Initialize pride positions and velocities
7: Initialize alpha, beta, and delta positions
8: while not converged do
9: Update positions and velocities of grey wolves based on GWO
10: Update positions and velocities of lions based on LOA
11: Calculate fitness values for grey wolves and lions
12: Update alpha, beta, and delta positions for both GWO and LOA
13: end while
14: Use the optimized parameters to train Hybrid CNN-LSTM Deep Learning model
15: return Model performance on testing data.

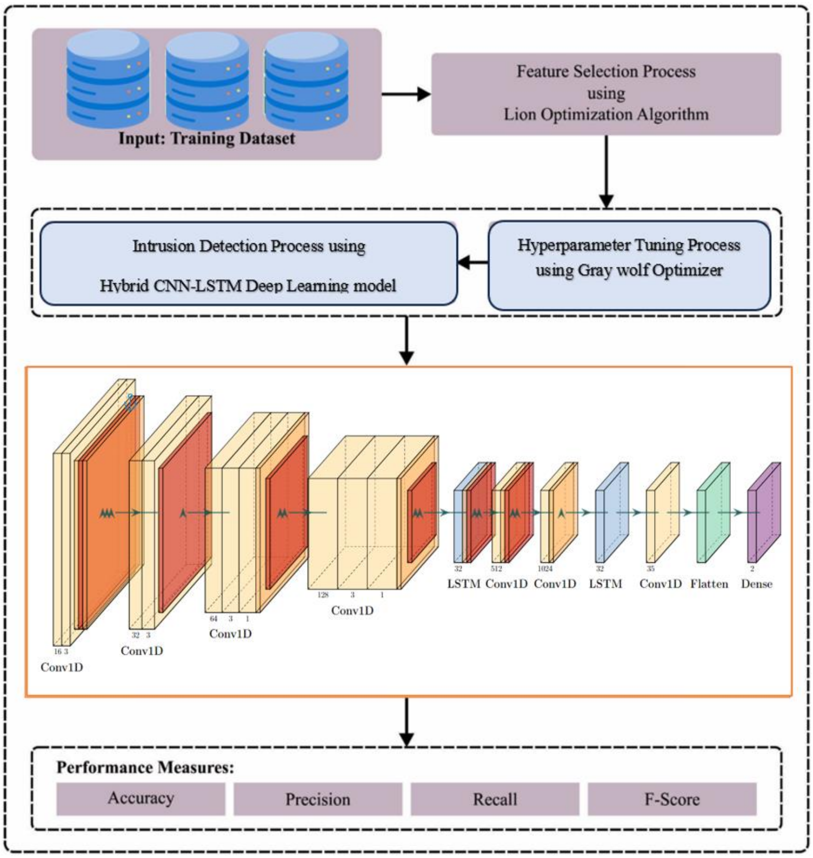

Figure 1. Proposed Intrusion Detection System

3.1 Dataset and data preprocessing

The data used in this study were obtained from the WUSTL-EHMS 2020 dataset using ARGUS [23]. The dataset comprises cases of MiTM attacks, including spoofing and data injection, to mimic probable security threats within an IoMT environment. Information stored in the CSV format contains network flow traffic and patient biometric data, as illustrated in Table 2.

Although the KDD'99 dataset was proposed to solve related issues, as defined in [23], the

NSL-KDD dataset was proposed as an alternative. However, it is recognized that some of the defects pinpointed in the NSL-KDD dataset by McHugh [24] can still be found and that it cannot possibly characterize real-world networks completely, which will augment the existing library of datasets as a helpful resource for researchers in the IDS field. One of the benefits of the NSL-KDD dataset is the removal of the redundancy present in the original KDD dataset, which was rampant with repeated records as illustrated in Tables 3 and 4. The insertion of repeated datasets into the KDD is one of the causes that predisposes learning algorithms to conservatism with respect to frequent records; thus, the detection of rare but susceptible attacks such as U2R and R2L is difficult. This problem is solved with the NSL-KDD dataset because it considers both frequent and infrequent patterns of network traffic, thereby improving the performance of the IDSs when learning from both dataset instances.

Table 2: WUSTL-EHMS 2020 Dataset Statistical Information [25]

Measurement	Value
Data size	4. 4 MB
Number of normal samples	14,272 (87. 5%)
Number of attack samples	2,046 (12.5%)
Total number of samples	16 ,318

Table 3: Statistics of redundant records in KDD train set [26]

	Original records	Distinct records	Reduction rate
Attacks	3,925,650	262,178	93. 32%
Normal	972,781	812,814	16. 44%
Total	4,898,431	1,074,992	78. 05%

Table 4: Statistics of redundant records in KDD test set [26]

	Original records	Distinct records	Reduction rate
Attacks	250,436	29,378	88. 26%
Normal	60,591	47,911	20. 92%
Total	311,027	77,289	75. 15%

Data preprocessing consists of many stages such as data cleaning, normalization and encoding. The handling of missing values, outliers, and appropriate encoding for categorical variables are considered. Normalization also ensures that the features are not out of scale, compromising the ability to dominate the model training.

3.2. Feature extraction and hyperparameter

Feature selection is the key to improving the model's ability to efficiently identify intrusions. The focus of this study was the second important feature of an algorithm to support multi-objective optimization. This feature is known as the Lion Optimization Algorithm (LOA) [27] and Grey Wolf Optimizer (GWO) [28] for the issue related to hyperparameter tuning. The process hinges on the lion optimization feature selection and grey wolf optimizer, which are used to obtain better results using the intrusion detection system [29]. LOFS is used for the identification of those features that have an impact on the network traffic data; however, GWO is applied for tuning the hyperparameters to explore the best possible performance of the proposed approach [30].

3.2.1 Lion Optimization Feature Selection (LOFS)

LOFS is a metaheuristic algorithm created based on a lion's symbiosis seen in real life. This interplay resembles the collective hunting technique of lions, whereby individual lions work together to increase hunting efficiency. The iteration LOFS is performed by a certain dataset subset of features to be selected based on their significance and importance contribution to the current task.

The population size for the Lion Optimization Algorithm (LOA) is taken as the default of 10 agents, and the number of iterations was set to 50. The loss function was defined as the binary cross-entropy function, and the model was trained for 10 epochs. This can be accomplished by encoding the augmentation method into the range of hyperparameters for LOA, which was between 0.0001 and 0.1. These parameters were perfectly adjusted for the system to detect possible intrusions into the invocation of all resources. The procedure is repeated until a stopping boundary is met, for instance, by completing a certain number of iterations or running the program to reach the desired output, as shown in Algorithm (2).

Algorithm 2: Lion Optimization Feature Selection (LOFS)
Input: Feature matrix X, Label vector y, Number of lions N, Maximum iterations MaxIter
Output: Selected feature

1: Initialize the pride of lions with random solutions
2: while not converged and not reaching maximum iterations do
   a: Evaluate the fitness of each lion in the pride based on classification perfor mance
   b: Identify the fittest lion as the leader (alpha)
   c: Update the position of each lion using the LOFS equations
3: end while
4: Select features based on the position of the leader lion
5: return Selected feature

Intrusion detection systems exploit the benefits of the LOFS method for feature selection. It not only reduces the data dimensions but also speeds the computations and maintains lower risks of overfitting. Undeniably, the reduction of less important features in the intrusion detection model by the LOFS process makes the model interpretable and user-friendly, in the sense that the features and patterns are easy to understand.

3.2.2 Grey Wolf Optimizer (GWO) Hyperparameter Tuning

The GWO borrowed its name from the behavior of social grey wolves in nature, which includes elements such as population-based metaheuristic optimization. In GWO, the population of alpha, beta, and delta wolves, headed by pack leaders, is equally divided into groups. These wolves indeed work together in a disparate population in an attempt to obtain the best answer to a given optimization problem. In the context of hyperparameter tuning for intrusion detection systems, reinforcement learning uses GWO to improve hyperparameter values, such as learning rates, regularization parameters, and network architecture parameters.

GWO's search space definition is performed based on the characteristics of the problem, which includes the feature selection to be analyzed. Each tuple specified a range of possible values for a particular feature. A pack size of 10 indicates that the GWO works with 10 wolves. Moreover, iterations, also known as loops, are critical features that determine the length of optimization and how well the algorithm approaches the best solution. The looping duration was set as 100. The main idea that underlies the GWO algorithm is the aim formulated for the health measure of reading each wolf's location in the search space. The main aim of GWO is to regularly update the wolves, As positions within the function that has to minimize errors until the goal of the optimal solution is finally found. With its refinement of the systematic parameters, it is possible to make the system operate at its full performance, thus enabling efficient classification and accurate detection.

This is done by systematically exploring the hyperparameter space and choosing the correct model configuration through the evaluation process indicated by the fitness function; thus, optimization of the intrusion detection system can be achieved as shown in Algorithm (3).

Algorithm 3: Grey Wolf Optimizer (GWO) Hyperparameter Tuning

Input: Objective function f (x), Number of wolves N, Maximum iterations MaxIter

Output: Optimized hyperparameters

1: Initialize the positions of grey wolves randomly

2: while not converged and not reaching maximum iterations do

a: Calculate fitness of each grey wolf based on objective function

b: Identify the alpha, beta, and delta wolves (leaders)

c: Update positions of each grey wolf using the GWO equations

3: end while

4: return Optimized hyperparameters

The interaction between the LOA and GWO empowers the suggested algorithm to accurately determine the most important features from the input data and simultaneously optimize the model's parameters for intrusion detection. Through this interconnected approach, the intrusion detection system was improved in terms of reliability, effectiveness, and resistance against various cyber threats that might occur in real life. Hence, this system is confident and capable of detecting and neutralizing these threats in real-world conditions.

3.3 Model architecture

The architecture of the Fast Hybrid Deep Learning Model is at the center of the proposed approach. The CNN-LSTM layers in Figure 2 and Table 5 are designed to extract complex patterns from the preprocessed data. Convolutional layers are followed by max-pooling layers and LeakyReLU to help in extracting the various features in a hierarchal manner. Some of the denser layers close to the edge of the architecture finely tuned these characteristics to give an output for intrusion detection.

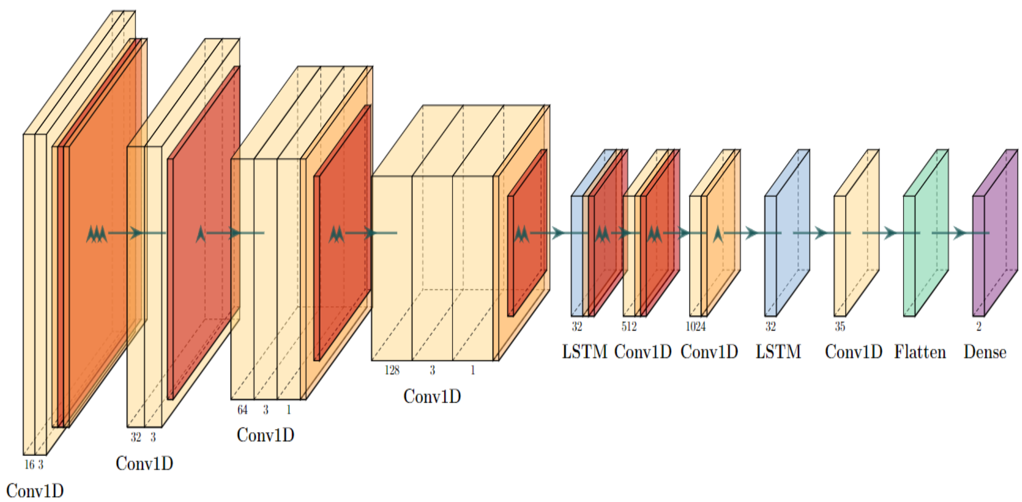

Figure 2. Hybrid deep-learning model

Table 5: Hybrid deep learning model layers

Layer	Parameters	Output Shape
Conv1D-1	Filters: 16, Kernel: 3, Stride: 1	(None, 40, 16)
LeakyReLU-1	-	(None, 40, 16)
MaxPool1D-1	Pool Size: 2	(None, 20, 16)
LeakyReLU-2	-	(None, 20, 16)
Conv1D-2	Filters: 32, Kernel: 3, Stride: 1	(None, 20, 32)
MaxPool1D-2	Pool Size: 2	(None, 10, 32)
Conv1D-3	Filters: 64, Kernel: 3, Stride: 1	(None, 10, 64)
LeakyReLU-3	-	(None, 10, 64)
MaxPool1D-3	Pool Size: 2	(None, 5, 64)
Conv1D-4	Filters: 128, Kernel: 3, Stride: 1	(None, 5, 128)
LeakyReLU-4	-	(None, 5, 128)
MaxPool1D-4	Pool Size: 2	(None, 3, 128)
LSTM	Units: 32	(None, 3, 32)
LeakyReLU-5	-	(None, 3, 32)
MaxPool1D-5	Pool Size: 2	(None, 2, 32)
Conv1D-5	Filters: 512, Kernel: 3, Stride: 1	(None, 2, 512)
LeakyReLU-6	-	(None, 2, 512)
MaxPool1D-6	Pool Size: 2	(None, 1, 512)
Conv1D-6	Filters: 1024, Kernel: 3, Stride: 1	(None, 1, 1024)
LeakyReLU-7	-	(None, 1, 1024)
LSTM-2	Units: 32	(None, 1, 32)
Conv1D-7	Filters: 35, Kernel: 3, Stride: 1, Activation: linear	(None, 1, 35)
Flatten	-	(None, 35)
Dense	Units: 2, Activation: softmax	(None, 2)

Thus, to sum up this hybrid CNN-LSTM design proposal, which comprises many layers, each layer should be designed as a feature extractor with the additional aim of constructing temporal dependency of sequences inherent in network analysis data for more accurate intrusion detection.

A multilayered neural network model mainly designed the model planning to enhance the ability of intrusion detection. Conv1D-1 is named to be the first layer, which is 1D convolution layer which has 16 filters, kernel size of three and the stride length to be one and the padding at both ends is made equal. This layer gets features using the received signals or input data from the preceding layer. After that, there is LeakyReLU-1, followed by the activation layer into which we introduce non-linearity so that the model can capture the complex patterns presented to it. Next would be the MaxPool1D-1; which is mandated with reducing the spatial dimensions by half, this reduces the computational load but only 'listens' to those feature maps distilled out by this process.

The next architecture named Conv1D-2 is also the same procedure with 32 filters and similar characteristics, followed by another LeakyReLU and MaxPool. This sequence is useful in making the network more abstract with the features learnt as it goes deeper in layers. In Conv1D_3 and Conv1D_4 abstraction of filters is taken to the next level with 64 and 128 filters respectively, with LeakyReLU and MaxPool accompanied each of them.

The LSTM layer that follows needs to be introduced after the preceding one, using 32 units as a necessary condition. This is helpful in capturing time-oriented dependencies of data well, since sequence-oriented intrusion detection is crucial. Further, to scale down the features, another LeakyReLU activation along with the MaxPooling layers is incorporated in the model.

Two more convex layers are introduced, namely Conv1D-6 and Conv1D-7, in the architecture above. Each one of these has 512 and 1024 filters individually and are accompanied by a LeakyReLU activation adding significantly to the layer complexity, thus enabling the network to learn quite complex patterns easily. Then added a second LSTM layer comprising 32 units for additional temporal analysis. Finally, the last Conv1D-8 layer comprises 35 filters with activation function other than linear, that readies the data for the dense output layer.

The conversion of the 3D tensor into a single vector is done by the Flatten layer. This vector is then passed on to a Dense layer that has only two units and whose activation function is SoftMax; it is through this layer that we eventually get our classification result. The system, with this layered architecture, can be said to effectively handle intricately patterned data which in turn makes the intrusion detection system more accurate and reliable.

4. Evolution measures

Performance measures are important for assessing how well different algorithms perform in detecting malicious code. The following standard metrics were employed in the evaluation process [31]:

1. True Positive (TP): This means true false positives.
2. True Negative (TN): This means the identification of innocent code cases.
3. False Positive (FP): This happens when a detector falsely labels benign files as malware.
4. False Negative (FN): It is the case when a detector fails to detect malicious code instances, particularly new viruses where no signature exists.

These metrics are encapsulated in the Confusion Matrix, which includes the following key terms:

1. Accuracy: The Reality accuracy of correct classification marking is achieved by using Eq. (1).
2. Precision: This is the value that shows how good the positive evaluations may be against positive True Positives (TP) as in Eq. (2).
3. Recall: The number of actual positive instances in total divided by the number of positively expected cases is contained in Eq. (3).
4. F-measure: F-measure (which is Precision plus Recall divide by (2 × Precision × Recall)) is a recall and precision weighted average that is often used for tasks involving text classification and question answering as illustrated in Eq. (4).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

$$F_1 = 2 * \frac{precision * recall}{precision + recall} = \frac{2TP}{2TP + FP + FN} \quad (4)$$

These performance measurements therefore serve as an exhaustive evaluation framework within which the methods of evaluation if these algorithms can perform to detect correctly with high accuracy, precision, recall and overall effectiveness can be analyzed. Apart from these statistics, the Confusion Matrix also gives enough insight into the manner how well the algorithm classifies between benign and malicious cases correctly.

5. Result and discussion

Practically, many parameters and processes appear as highly significant in determining the performance of the proposed model. We delved deep into this research study, where we applied fine-tuned hyperparameters for the deep learning models and Grey Wolf Optimizer (GWO) algorithm aiming at enhancing Intrusion Detection System (IDS) performance. The key hyperparameters considered in our deep learning models included the number of layers, learning rate, batch size, and epochs. Our CNN and LSTM layers' depths were experimented with different ranges from 3 to 7 layers to find an optimal value. We looked into the impact of the learning rate set at 0.001 on convergence speed and accuracy, as well as balancing computational efficiency against model performance using batch sizes of either 32, 64, or 128. While also making sure we did not compromise on underfitting or overfitting by varying the training epochs through 50, 100, and up to 150 counts. An ablation study was conducted where we systematically checked each hyperparameter's contribution. We discovered that a 7-layer network with a learning rate of 0.001, batch size of 64 and training over 100 epochs gave us the best results for our model using GWO algorithm. The study focused on tuning key hyperparameters the number of wolves (agents), iterations and search space dimension by ensuring an adequate exploration and exploitation of the search space: the number of wolves varied between 10, 20, and 30 while the number of iterations took values of either 100, 200, or 300. The search space dimension (reflecting the number of features) was adjusted between 5, 10, and 15 as well during this tuning process. It was found that the optimal performance is achieved when employing 20 wolves, 200 iterations, and a 10-dimensional search space in the Grey Wolf Optimizer balancing the quality of solutions and computational cost. We could achieve significant enhancements in our IDS performance through these hyperparameter optimizations without neglecting detailed ablation studies, where such information allows us to get important clues about behavior of model-algorithmic systems.

An illustration of the training process presents a lift in training accuracy on iterative basis, eventually attaining a flatline at around 99.26%. It can be seen that it is concluded that trained model not only learns from the data because the high accuracy is not obtained by overfitting. In contrast, one of the curves displays a constant reduction with each successful iteration of training, which reflects the fact that the better the model becomes at predicting the desired results, the lower the errors that it makes. As the trend of decreasing loss goes forward, it indicates that the model is minimizing the difference between outputs and inputs, and it is also optimizing the model's parameters as shown in Figures 3 and 4. Altogether, those plots show a trained model at this level which is the one with high accuracy and low error, therefore it is good in pattern learning from the data and prediction making.

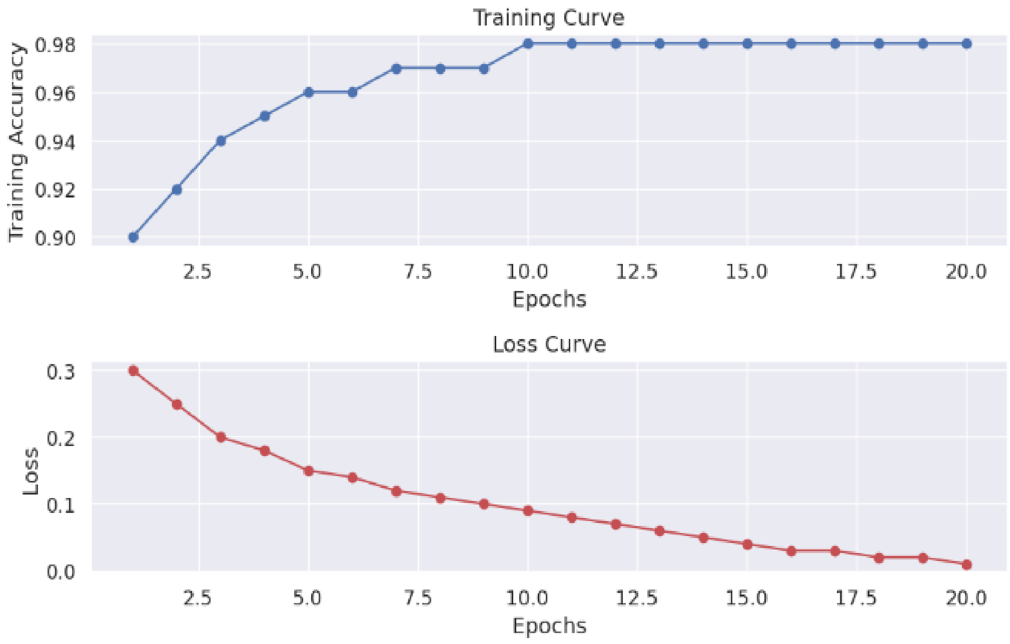

Figure 3. Training and loss curves

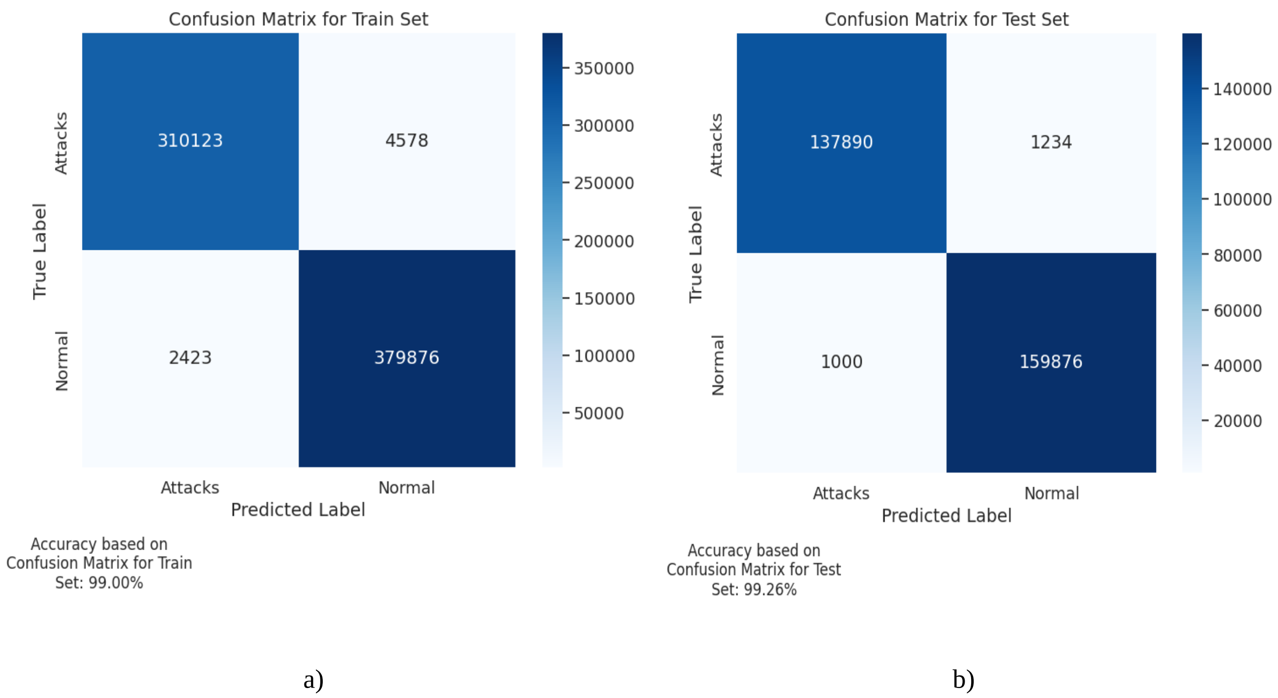

Figure 4. Confusion Matrices, a) Accuracy based on Train set, b) Accuracy based on Test set

Through this comparative analysis by Figures 5 and 6 the research is able to evaluate various intrusion detection approaches including LOFSGWO, BBAFS-DRL, SVM, NB-Bagging, NB-Adaboost, GCNSE, and CNN-Adaboost, CSTR, and ELM-SVM in terms of their capabilities in detecting network anomalies. Hence the LOFSGWO reveals the best performance in terms of all measures, obtains the accuracy of 99.26. This is a great sign that suggested LOFSGWO method has such capacity of detecting both the attack and normal instance. Rather than that, the competition performance tends to fluctuate between very good or poor from one approach to another including BBAFS-DRL, SVM, NB-Bagging, NB-Adaboost, GCNSE, and CNN-Adaboost. BBAFS-DRL, however, can be considered as a successful approach to animate our character as it outputs an accuracy of 95.04% and F1 score of 95.04%. Both these values are slightly lower than those achieved by the LOFSGWO but not to a great extent. By contrast to that, the SVM presents less accuracy and F1 score amounting to 75.91% and 77.76%, which is lower than the proposed LOFSGWO method's FAPP. Concurrently, NB-Bagging and NB-Adaboost often tend to score 70.01% and 71.34% in terms of accuracy and F1 score correspondingly with 70.93% and 74.07% values respectively.

GCNSE performs with the higher accuracy and F1 score than the SVM, NB-Bagging, and NB-Adaboost, regarding values of accuracy and F1 score are 80.17%, 80.82% accordingly. Moreover, it seems to have some weaknesses compared to the List of Other Frequently Spoken Languages in the World. To sum up, CNN-Adaboost reveals the lowest accuracy of 74.16% and the worst F1 score of 68.16% as compared to other options aimed to address the issue, thus it might be called the worst performer with respect to other methods. The more confusing the matrices are, the better performance classes is assigned to the different methods. LOFSGWO shows a high number of correctly classified cases with particularly strong performance for attacks, as revealed in the high values in both the confusion matrices for correct (true) classes. Similarly, the application of other techniques including SVM and CNN-Adaboost lead to less efficient performance while with more errors discovered in misclassifications

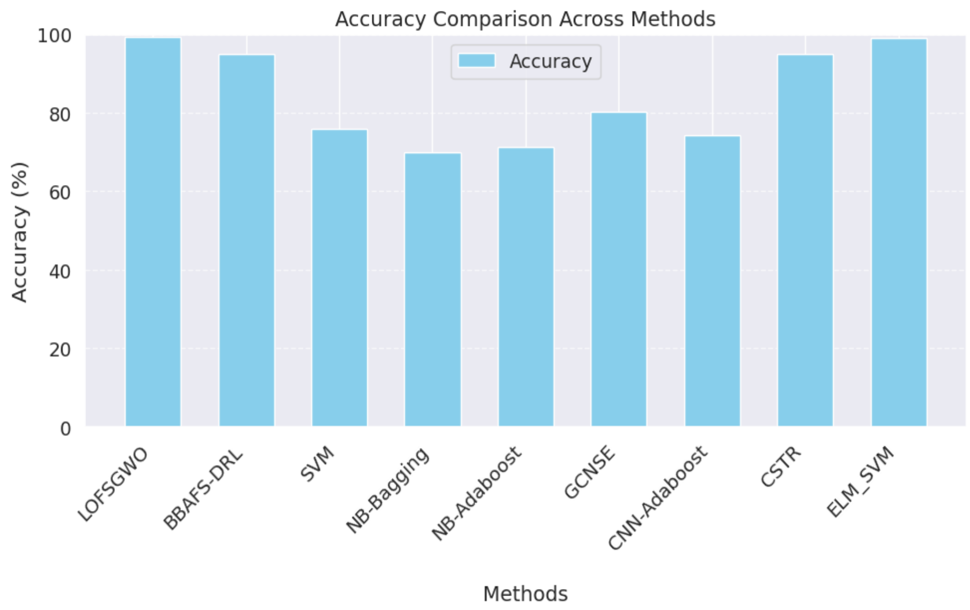

Figure 5. Model accuracy performance metrics with [21,22], [32,33].

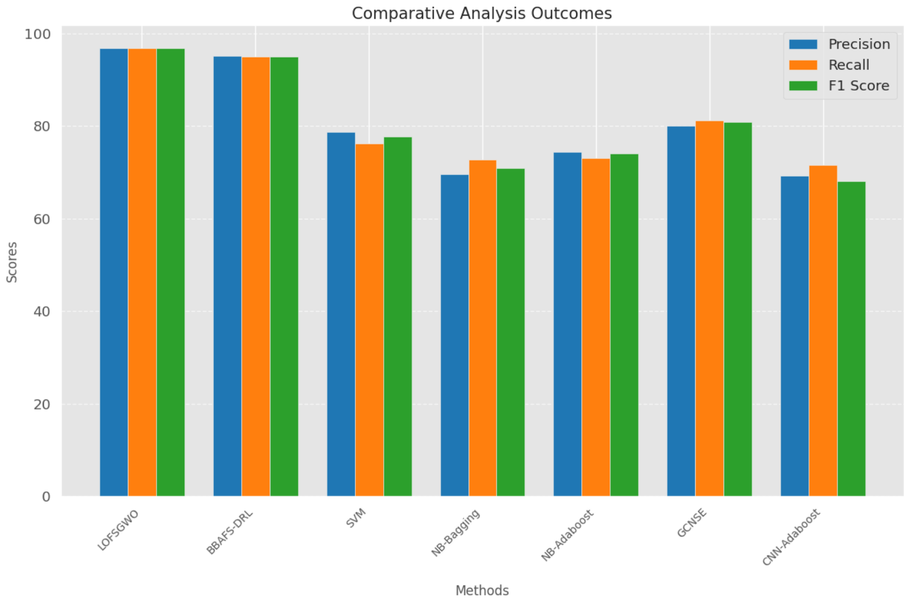

Figure 6. Model performance metrics with [32,33]

6. Conclusions

The system design initially comes to the idea of the combination of Lion Optimization Selection and Grey Wolf Optimizer for hyperparameter optimization of IDS will serve as a possible way to boost accuracy and efficiency. Developing Hybrid CNN-LSTM algorithm with LOFS and GWO alternatives produced the highest accuracy of prediction at 99.26% which exceeded well known algorithms now used worldwide. Further, the results demonstrated the most improvements regarding precision, recall, and F1 score metrics, proving that the system operates in very fast response times, even in the case of different network traffic anomalies. These results speak of inability of metaheuristic algorithm in improving the IDS performance. Therefore, the attributes selection and the hyperparameter tuning must be applied if better detection capabilities are to be achieved.

By this, future studies are recommended to focus on deeper sensing and understanding psychological pressure in the context of online environmental groups. This mechanism should be complemented by provisions addressing divergent types of networking environments, thus making the system more accommodative and effective. Next to that, the combination of several algorithms or a hybrid approach with increase the outcome as well. In addition, we want to look into those emerging threats and new technologies, such as IoT and cloud, which are rapidly spreading throughout the networks. These present great fields to cover in future explorations. Sustaining the commitment to broad research in this field may curtail network vulnerabilities from all angles, and hence, withstand adversarial cyber threats of today's digital-driven world.

References

[1] A. Thakkar and R. Lohiya, “Fusion of statistical importance for feature selection in Deep Neural Network-based Intrusion Detection System,” Information Fusion, vol. 90, pp. 353–363, Feb. 2023.

[2] M. B. Pranto, M. H. A. Ratul, M. M. Rahman, I. J. Diya, and Z. bin Zahir, “Performance of Machine Learning Techniques in Anomaly Detection with Basic Feature Selection Strategy-A Network Intrusion Detection System,” Journal of Advances in Information Technology, vol. 13, no. 1, 2022.

[3] I. Katib and M. Ragab, “Blockchain-Assisted Hybrid Harris Hawks Optimization Based Deep DDoS Attack Detection in the IoT Environment,” Mathematics, vol. 11, no. 8, p. 1887, Apr. 2023.

[4] M. Moizuddin and M. V. Jose, “A bio-inspired hybrid deep learning model for network intrusion detection,” Knowledge-Based Systems, vol. 238, p. 107894, Feb. 2022.

[5] M. Ahsan, R. Gomes, M. M. Chowdhury, and K. E. Nygard, "Enhancing Machine Learning Prediction in Cybersecurity Using Dynamic Feature Selector," Journal of Cybersecurity and Privacy, vol. 1, no. 1, 2021.

[6] M. A. Talukder et al., "A dependable hybrid machine learning model for network intrusion detection," Journal of Information Security and Applications, vol. 72, 2023.

[7] G. Sah, S. Banerjee, and S. Singh, "Intrusion detection system over real-time data traffic using machine learning methods with feature selection approaches," International Journal of Information Security, vol. 22, no. 1, pp. 1–27, Oct. 2022.

[8] M. Maabreh, I. Obeidat, E. A. Elsoud, A. Alnajjai, R. Alzyoud, and O. Darwish, "Towards Data-Driven Network Intrusion Detection Systems: Features Dimensionality Reduction and Machine Learning," International Journal of Interactive Mobile Technologies, vol. 16, no. 14, 2022.

[9] M. Ragab, S. M. Alshammari, and A. S. Al-Malaise Al-Ghamdi, "Modified Metaheuristics with Weighted Majority Voting Ensemble Deep Learning Model for Intrusion Detection System," Computer Systems Science and Engineering, vol. 47, no. 2, 2023.

[10] M. Ragab and M. Farouk S. Sabir, "Outlier detection with optimal hybrid deep learning enabled intrusion detection system for ubiquitous and smart environment," Sustainable Energy Technologies and Assessments, vol. 52, 2022.

[11] G. Kocher and G. Kumar, "Analysis of Machine Learning Algorithms with Feature Selection for Intrusion Detection using UNSW-NB15 Dataset," International Journal of Network Security & Its Applications, vol. 13, no. 1, pp. 21–31, Jan. 2021.

[12] B. Sharma, L. Sharma, C. Lal, and S. Roy, "Anomaly based network intrusion detection for IoT attacks using deep learning technique," Computers and Electrical Engineering, vol. 107, p. 108626, Apr. 2023.

[13] 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, Feb. 2023.

[14] S. Hosseini and S. R. Sardo, "Network intrusion detection based on deep learning method in internet of thing," Journal of Reliable Intelligent Environments, vol. 9, no. 2, pp. 147–159, Feb. 2022.

[15] N. F. Syed, M. Ge, and Z. Baig, "Fog-cloud based intrusion detection system using Recurrent Neural Networks and feature selection for IoT networks," Computer Networks, vol. 225, 2023.

[16] M. L. Hernandez-Jaimes, A. Martinez-Cruz, K. A. Ramírez-Gutiérrez, and C. Feregrino-Uribe, "Artificial intelligence for IoMT security: A review of intrusion detection systems, attacks, datasets and Cloud–Fog–Edge architectures," Internet of Things (Netherlands), vol. 23. 2023.

[17] C. Chen, Y. Gao, S. Huang, and X. Yan, "Avoid attacks: A Federated Data Sanitization Defense in IoMT Systems," IEEE INFOCOM 2023 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), May 2023.

[18] A. Binbusayyis, H. Alaskar, T. Vaiyapuri, and M. Dinesh, "An investigation and comparison of machine learning approaches for intrusion detection in IoMT network," The Journal of Supercomputing, vol. 78, no. 15, pp. 17403–17422, May 2022.

[19] Y. K. Saheed and M. O. Arowolo, "Efficient Cyber Attack Detection on the Internet of Medical Things-Smart Environment Based on Deep Recurrent Neural Network and Machine Learning Algorithms," IEEE Access, vol. 9, 2021.

[20] Y. Manchala, J. Nayak, and H. S. Behera, "Detection of Malicious Traffic in IoMT Environment Using Intelligent XGboost Approach," 2022 OPJU International Technology Conference on Emerging Technologies for Sustainable Development (OTCON), Feb. 2023.

[21] J. Jithish, S. Sankaran, and K. Achuthan, "A Hybrid Machine Learning Approach for Intrusion Detection in Cyber-Physical Manufacturing Systems," Intelligent Security Solutions for Cyber-Physical Systems, pp. 156–168, Mar. 2024.

[22] E. M. Maseno and Z. Wang, "Hybrid wrapper feature selection method based on genetic algorithm and extreme learning machine for intrusion detection," Journal of Big Data, vol. 11, no. 1, Feb. 2024.

[23] A. Hady, A. Ghubaish, T. Salman, D. Unal, and R. Jain, "Intrusion Detection System for Healthcare Systems Using Medical and Network Data: A Comparison Study," IEEE Access, vol. 8, pp. 106576–106584, 2020.

[24] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in IEEE Symposium on Computational Intelligence for Security and Defense Applications, CISDA 2009, 2009.

[25] A. Ghubaish, "WUSTL EHMS 2020 Dataset for Internet of Medical Things (IoMT) Cybersecurity Research," Wustl.edu, 2020. https://www.cse.wustl.edu/~jain/ehms/index.html

[26] "NSL-KDD | Datasets | Research | Canadian Institute for Cybersecurity | UNB," www.unb.ca. https://www.unb.ca/cic/datasets/nsl.html

[27] J. Mchugh, "Testing Intrusion Detection Systems: A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations as Performed by Lincoln Laboratory," ACM Transactions on Information and System Security, vol. 3, no. 4, 2000.

[28] M. Yazdani and F. Jolai, "Lion Optimization Algorithm (LOA): A nature-inspired metaheuristic algorithm," Journal of Computational Design and Engineering, vol. 3, no. 1, pp. 24–36, Jun. 2015.

[29] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," Advances in Engineering Software, vol. 69, 2014.

[30] H. M. Fadhil, M. N. Abdullah, and M. I. Younis, "TWGH: A Tripartite Whale–Gray Wolf–Harmony Algorithm to Minimize Combinatorial Test Suite Problem," Electronics, vol. 11, no. 18, p. 2885, Sep. 2022.

[31] H. M. Fadhil, N. Q. Makhool, M. M. Hummady, and Z. O. Dawood, "Machine Learning-based Information Security Model for Botnet Detection," Journal of Cybersecurity and Information Management (JCIM), vol. 9, no. 1, pp. 68–79, 2022.

[32] A. Wang, W. Wang, H. Zhou, and J. Zhang, "Network Intrusion Detection Algorithm Combined with Group Convolution Network and Snapshot Ensemble," Symmetry, vol. 13, no. 10, pp. 1814, Sep. 2021.

[33] S. Priya and K. P. M. Kumar, "Binary bat algorithm-based feature selection with deep reinforcement learning technique for intrusion detection system," Soft Computing, vol. 27, no. 15, pp. 10777–10788, Jun. 2023.