Computers, Materials & Continua
DOI: 10.32604/cmc.2023.033153
Article

Hybrid Grey Wolf and Dipper Throated Optimization in Network Intrusion Detection Systems

Reem Alkanhel[1,*], Doaa Sami Khafaga[2], El-Sayed M. El-kenawy[3], Abdelaziz A. Abdelhamid[4,5], Abdelhameed Ibrahim[6], Rashid Amin[7], Mostafa Abotaleb[8] and B. M. El-den[6]

[1]Department of Information Technology, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh, 11671, Saudi Arabia
[2]Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh, 11671, Saudi Arabia
[3]Department of Communications and Electronics, Delta Higher Institute of Engineering and Technology, Mansoura, 35111, Egypt
[4]Department of Computer Science, Faculty of Computer and Information Sciences, Ain Shams University, Cairo, 11566, Egypt
[5]Department of Computer Science, College of Computing and Information Technology, Shaqra University, 11961, Saudi Arabia
[6]Computer Engineering and Control Systems Department, Faculty of Engineering, Mansoura University, Mansoura, 35516, Egypt
[7]Department of Computer Science, University of Engineering and Technology, Taxila, Pakistan
[8]Department of System Programming, South Ural State University, Chelyabinsk, 454080, Russia
*Corresponding Author: Reem Alkanhel. Email: rialkanhal@pnu.edu.sa
Received: 09 June 2022; Accepted: 01 August 2022

Abstract: The Internet of Things (IoT) is a modern approach that enables connection with a wide variety of devices remotely. Due to the resource constraints and open nature of IoT nodes, the routing protocol for low power and lossy (RPL) networks may be vulnerable to several routing attacks. That's why a network intrusion detection system (NIDS) is needed to guard against routing assaults on RPL-based IoT networks. The imbalance between the false and valid attacks in the training set degrades the performance of machine learning employed to detect network attacks. Therefore, we propose in this paper a novel approach to balance the dataset classes based on metaheuristic optimization applied to locality-sensitive hashing and synthetic minority oversampling technique (LSH-SMOTE). The proposed optimization approach is based on a new hybrid between the grey wolf and dipper throated optimization algorithms. To prove the effectiveness of the proposed approach, a set of experiments were conducted to evaluate the performance of NIDS for three cases, namely, detection without dataset balancing, detection with SMOTE balancing, and detection with the proposed optimized LSH-SOMTE balancing. Experimental results showed that the proposed approach outperforms the other approaches and could boost the detection accuracy. In addition, a statistical analysis is performed to study the significance and stability of the proposed approach. The conducted experiments include seven different types of attack cases in the RPL-NIDS17 dataset. Based on the

proposed approach, the achieved accuracy is (98.1%), sensitivity is (97.8%), and specificity is (98.8%).

Keywords: Metaheuristics; grey wolf optimization; dipper throated optimization; dataset balancing; locality sensitive hashing; SMOTE

1 Introduction

Internet of Things (IoT) is a new trend that allows to a variety of sources to communicate and connect over the internet [1]; everything can communicate with anyone at any time. The IoT is made up of objects, also known as "things," and is equipped with smart devices with built-in intelligence that provides the ability to connect to each other and share data [2]. Intelligent devices exchange data for making decisions and executing operations. Internet protocol version 6 (IPv6) supports this connection by assigning a unique IP address to each intelligent device, thereby enabling it where the address can be specified globally [3,4]. IoT enables many applications that improve human life, but with many advantages. Due to its global linkage, accessible environment, and the limited resources of smart systems and wireless networks, the IoT is vulnerable to numerous routing attacks [5,6]. The long-term operation of many devices demands limited power usage in the network. The low-power wireless personal area network (6LoWPAN), based on IPv6, is an IoT network that allows IoT systems to function on minimal power [7]. 6LoWPAN networks do not support traditional IoT routing protocols due to the loss and minimal power nature of 6LoWPAN networks. In order to enable effective routing in 6LoWPAN networks, the RPL network was introduced [8]. The main advantages of the control messages are following a specific pattern during the repair and building an IoT network. RPL-based IoT networks offer Significant benefits in routing because of their mobility and battery life; it is subject to multiple routing attacks, including civils, selective forwarding, sinkholes, clone ID, black holes and local repair attacks. Apart from that, an attacker could compromise your privacy gaining user security and access through eavesdropping and financial loss of personal information. As a result, a strategy is needed to secure the IoT from different routing attacks. Many researchers have employed the NIDS to overcome this problem.

Anomaly-based NIDS (ANIDS) takes deviations from normal traffic attitude and network traffic as the basis for detecting attacks. ANIDS performance depends primarily on the analytical model's effectiveness (i.e., Training dataset classifiers and their reliability) [9]. Therefore, using machine learning (ML) algorithms to build effective ANIDS models and detect possible attacks in the IoT is the best solution. In IoT networks, normal and anomalous information is being utilized to learn ML classifiers for discovering attack. Several of the IDS research focuses on various classification techniques such as deep learning [10,11], ML [12,13], and ensemble learning [14–16]. The ensemble learning merges multiple classifiers to reduce error rates and achieve better classification. Several investigations have found that IDS classifier optimization algorithms, including black widow optimization, Tunicate swarm algorithms, genetic algorithms (GA), gray wolf optimization, particle swarm optimization (PSO), and salp swarm algorithms (SSA) are examples of metaheuristic based optimization algorithms that are widely used to improve the performance of machine learning models.

The dataset quality used for training is important because the error rate and accuracy of the classifier depend on it. As a result of datasets unbalancing, the classification task misclassifies minority classes. As a result, reduce the gap between the dominant and minority classes, is required to oversample the minority class. For balancing datasets, the SMOTE is the most efficient sampling technique. To overcome previous NIDS difficulties such as high tackle false alarm rate, unbalanced

dataset is balanced using a novel approach based on a new metaheuristic optimization algorithm base on grey wolf optimization and dipper throated optimization algorithms to obtain higher attack detection rates. An analysis of the assessment of the effectiveness is carried out to demonstrate that the proposed approach outperforms existing approaches. As a result, the proposed approach could increase the accuracy of hacks detection for the seven most prevalent attacks that occur in RPL-based IoT with a high detection rate.

The remainder of this paper is organized as follows: Related work in RPL-based NIDS using bio-inspired techniques is described in Section 2. The proposed methodology is then described in Section 3. Section 4 discusses the experimental results. Finally, in Section 5, the main conclusions of this work are presented.

2 Literature Review

The NIDS for RPL-based IoT, considering feature selection and ensemble classification procedures, are described in this part. The ensemble classifier has been used as an IDS in a number of publications. Authors in the literature have studied ML algorithms for detecting denial-of-service attacks (DoS) in the IoT. They have various ensemble classifiers such as AdaBoost, extreme gradient boosting, gradient boosting machines, random forest (RF) and extremely random Trees, and single classifiers like regression trees, classification, and multilayer perceptron. The used ensemble classifier achieved high praise 96.7% accuracy. Ensemble-based classifiers for RPL-based IoT networks were presented to detect seven types of routing threats. Bagged tree, boosted tree, random under-sampling (RUS)-boosted tree, and subspace discriminant boosted Trees are among the four classifiers included. Evaluate performance by evaluating various measurements like accuracy, area under the curve, and receiver operating characteristic (ROC) curve. The dataset RPL-NIDS17 is applied to train the classifier. With a detection accuracy equal to 94.5%, this ensemble technique was successful. However, they do not use classifier weight and feature selection optimization techniques to minimize computational complexity. Authors in [17] reported that an ensemble deep learning-based IDS using a deep neural network for IoT and a DT classifier employed in an industrial management platform had a detection rate of 95.86% [18]. Comparing different kinds of boosting algorithms like modest Ada-Boost, gentle Ada-Boost, and real Ada-Boost for NIDS evaluation by authors in [19] observed that the error rate for ensemble among these classifiers is reasonable in the region of 70%. It was evaluated the ability of ML-based IDSs utilizing a feature selection method known as the XG-boost algorithm. This approach used multiple classifiers such as support vector machines (SVM), linear regression (LR), decision trees (DT) and K-nearest neighbors (KNN) to examine the network attacks. The ensemble classifier has a detection accuracy equal to 90.85% [20]. Parallel quadratic ensemble learning based on IDS gradient strengthening DT is introduced to get more accurate detection of attacks such as distributed DoS, benign, port scans, and web attack traffic.

The optimum combination of feature selection and voting-based ensemble learning is discussed in some references. A suggested IDS using ensemble learning and feature selection approaches proposed by authors in [21]. Use hybrid correlation-based feature selection and the bat Algorithm (CFS-BA) approach to minimize the dimensions of information. After that, the penalizing algorithms introduced a voting ensemble method that combines random forest (RF), C4.5, and forest. Dataset is applied in the experimental evaluation to demonstrate the efficiency of the CFS-BA ensemble approach. Reduction of the modeling time for this model compared to the full feature model was achieved by using the feature selection technique. In addition, their method has a detection rate of 99.9% and better performance than other approaches. Implemented vote casting classifier and feature selection

strategies to locate the botnet attacks by authors in [22]. The PSO technique is utilized to identify the most significant features from the dataset. Deep neural network, SVM, DT, and C4.5 algorithms are used in vote systems to detect botnet attacks. Bot-IoT and ISOT datasets were used for this. They were successful in accomplishing a detection accuracy equal to 99.9%. The feature selection technique applied a hybrid of optimization techniques, i.e., ant colony, GA, and PSO. The two-phase of the ensemble is comprised of a Meta classifier and a base classifier, both of which are Meta classifiers. The proposed version is decreased the training time with the optimized feature selection strategy. They have an accuracy of 85.8% when it comes to detection. Cyber-attack protection mechanism for the internet of medical things is presented by authors in [23], based on an ensemble learning and fog-cloud design. Naïve bayes (NB), RF, and DT classifiers can be used to form ensemble learning models. Detecting attacks is achieved by sending the output of the classifier to the XG-Boost model to perform classification. This results in a false acceptance rate (FAR) reduction up to 5.59%, and achieved accuracy of 96.35%. Ensemble-based classifier with IDS features selection techniques is used in the previous work. However, no other research on RPL-based IoT with bio-inspired ensemble classification and hybrid feature selection has been published.

Several studies have offered different IDSs for IoT networks, but limited publications suggested IDSs for RPL-based IoT networks. Authors in [10] introduced a deep-learning-based gated recurrent unit (GRU) that can identify hello flooding (HF) attacks in RPL-based IoT networks with a high accuracy rate. The effectiveness of the suggested approaches is compared using SVM and LR classifiers. Many factors are used to evaluate the performance of this technique, including root mean square error, precision, mean absolute error, delay, usage of energy, and packet delivery rate. The detection accuracy of the GRU-based deep learning model is 99.96%, but only one attack can be detected. Authors in [24] devised a Gini coefficient-based countermeasure to preserve RPL-based networks from Sybil attacks. To find result improvement in terms of detection latency and detection rate, the proposed countermeasures are compared to two algorithms: two-phase detection and SecRPL. This method can achieve a detection rate of 100%, but it is unable to identify other sorts of attacks. Authors in [25] proposed a lightweight bio-inspired IDS based on artificial ones, and they deployed a bee colony to protect mobile RPL from Sybil attacks. According to specificity, traffic overhead control, accuracy, packet delivery rate, sensitivity, and usage of energy, the sybil attacks model was evaluated. This method has a 95% average detection accuracy and could only detect one sort of attack. Authors in [26] suggested a game model-based intrusion detection platform for protecting RPL-based networks. Two related methodologies such as evolutionary game to confirm attacks and a stochastic game to detect the attack. Various RPL attacks like local pairs, ranks, neighbors, and IDS attacks are detected by this method, but with lower detection accuracy and a false positive rate of 90% or more. The proposed approach could boost the detection accuracy by utilizing a new metaheuristic optimization approach to achieve a proper balance between the dataset categories of intrusion cases.

3 Methodology

The proposed network intrusion identification system is composed of a new bio-inspired LSH-SMOTE algorithm for dataset balancing in order to identify assaults in RPL-based IoT networks [27–33]. Fig. 1 depicts the proposed NIDS architecture. Data gathering, analysis, and detection are all part of the proposed work's design. Sensor events/traffic repositories and sniffers make up the data-collection unit. In the 6LoWPAN network, the sniffer is used to keep tabs on every packet that is sent. Sniffed sensor events and sent packets can be found in the repository of sensor events/traffic collection that is directly linked to it. Finally, the proposed optimized LSH-SMOTE approach is used to bring the dataset features into harmony. The fundamental aspect of the design is the analysis unit,

which is also known as the analysis engine. Detection includes an alarm/attack notification module that receives orders from the analysis unit to raise an alert if an attack is detected. It also analyzes the traffic periodically and provides log reports to the user interface, which stores all of the information.

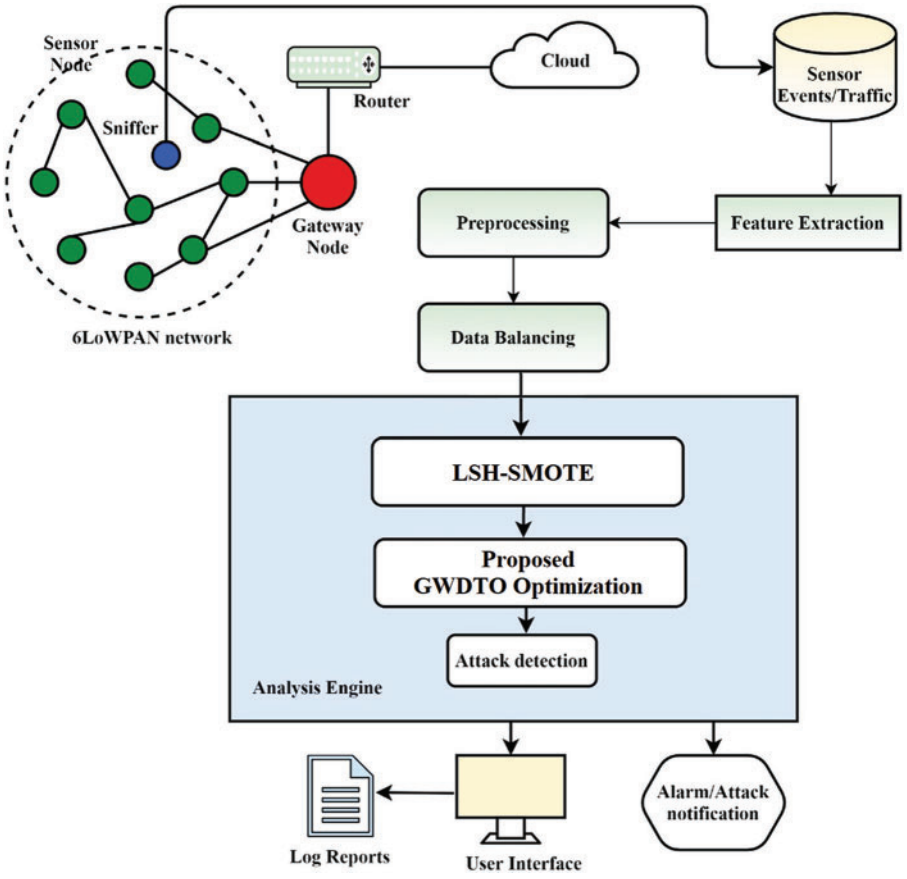

Figure 1: The proposed architecture of network intrusion detection

3.1 Dataset Collection

The proposed approach is trained on the RPL-NIDS17 dataset [34]. For the purpose of creating this dataset, the NetSim software was utilized. In a wide variety of network environment simulations, NetSim is a commonly used tool. To build the dataset, the IoT network consists of a gateway, sensor nodes, a wired node, and a router. Every assault is recorded in a distinct comma separated values (CSV) file, which contains all of the relevant information. A single dataset is created by merging all of the CSV files into one. There are two extra attributes for labeling in this dataset, making a total of 20 features containing time, basic, and flow features. The routing attack patterns include Sybil, blackhole, sinkhole, clone identifier (ID), local repair assaults, hello flooding, and selective forwarding in addition to the typical traffic pattern. There are 33,337 instances of routing attacks and 431,981 instances of regular traffic in this dataset. As a result, the data is unbalanced.

3.2 Dataset Preprocessing

An important initial step in the preparation of data is to clean it up. Removing duplicates, filling in any blanks, and encoding data are all part of this process. Only numeric data can be read by machines; nevertheless, the dataset consists of both numeric and nominal data. As a result, numeric values are generated from the dataset's characters via encoding. After all of this, data scaling is used to make things go faster. The dataset contains a wide range of features, both in terms of their size and their unit. If you want to preserve your data in a consistent manner, then scaling can help you achieve this goal [35–42].

3.3 Dipper Throated Optimization

Dipper throated optimization (DTO) is based on a simulation of the real process of looking for food by tracking the positions and speeds of swimming and flying birds. The following equations are used to update the position and speed of the swimming birds.

$$BL_{nd}(t+1) = BL_{best}(t) - C_1.|C_2.BL_{best}(t) - BL_{nd}(t)| \tag{1}$$

where $BL_{nd}(t)$ and $BL_{best}(t)$ are the normal location and best location of the bird at iteration t, and C_1 and C_2 are adaptive values whose values are changed during the optimization process based on the iteration number and random values. The update of the flying bird's location is performed using the following equation.

$$BS(t+1) = C_3 BS(t) + C_4 r_1 (BL_{best}(t) - BL_{nd}(t)) + C_5 r_1 (BL_{Gbest} - BL_{nd}(t)) \tag{2}$$

$$BL_{nd}(t+1) = BL_{nd}(t) + BS(t+1) \tag{3}$$

where $BS(t+1)$ is the updated speed of each bird, r_1 is a random number in [0; 1], BL_{Gbest} is the global best location, and C_3 is a weight value, C_4 and C_5 are constants.

3.4 Grey Wolf Optimization

An alpha, beta, or omega wolf is regarded as a subpar wolf in the optimization of the Grey Wolf (or, according to some references, delta). The omega is ruled by delta wolves, who are superior to the alphas and betas. Scouts, elders, and hunters make up this group. They rely on the hunters to help them search for food, and the hunters give the group with food as a result. The security of the organization is the responsibility of the Sentinels. The primary duty of a scout is to keep an eye out for any dangers to the group's area and notify the rest of the unit accordingly. The wolves that have served as alpha or beta in the past are known as the pack's Elders. Even more intriguing than the social organization of grey wolves is the way they hunt as a group. As previously reported, the grey wolves are encircling their prey. Encircling behavior may be modeled using the following equations:

$$\vec{F}(t+1) = \vec{F}_p(t) - \vec{A}.\vec{D} \tag{4}$$

$$\vec{D} = \left|\vec{C}.\vec{F}_p(t) - \vec{F}(t)\right| \tag{5}$$

where $\vec{A}$ and $\vec{C}$ are vectors of coefficients, t represents the current iteration, $\vec{F}$ is the grey wolf's position vector, and $\vec{F}_p$ indicates the position vector of the prey. If there is a better solution in each iteration, $\vec{F}$ is updated to the best solution.

$$\vec{a} = 2 - t\left(\frac{2}{Max_{iter}}\right) \tag{6}$$

$$\vec{A} = 2\vec{a}.\vec{r}_1 - \vec{a} \tag{7}$$

$$\vec{C} = 2\vec{r}_2 \tag{8}$$

Loop counter t, Max_{iter} is the maximum number of iterations, $\vec{r}_1$ and $\vec{r}_2$ are random vectors in [0,1], and $\vec{a}$ is linearly decreasing from 2 to 0 throughout the length of iterations. Consider a two-dimensional position vector and some of the potential neighbors depicted in Fig. 2 as a starting point for exploring the impact of Eqs. (1) and (2).

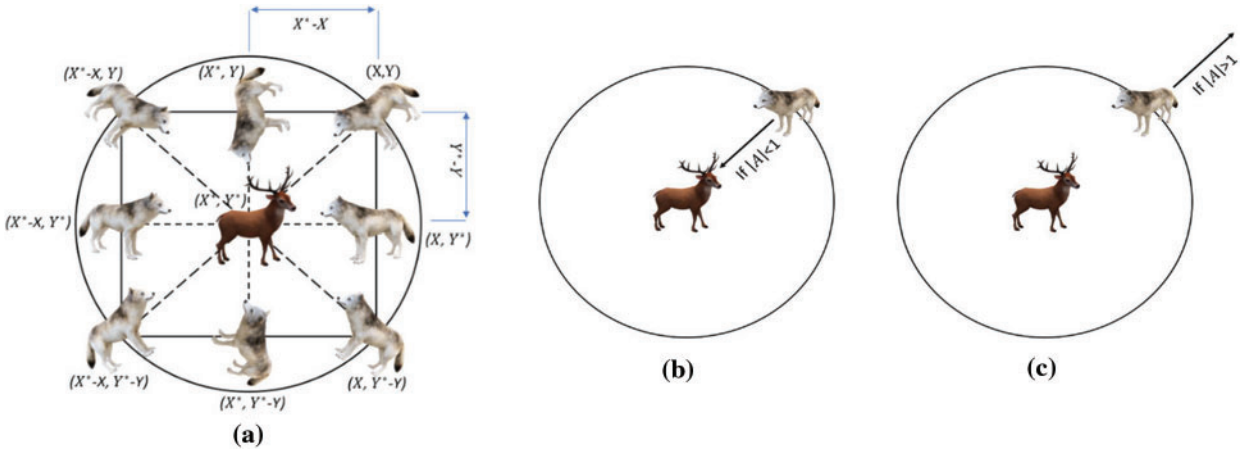

Figure 2: Hunting process: (a) potential locations of grey wolves (b) convergence (attacking the prey), (c) divergence (searching for prey)

Algorithm 1: The proposed DTO-GW algorithm

Initialize birds locations $BL_i(i = 1, 2, 3, \ldots, n)$ with size n, $BS_i(i = 1, 2, 3, \ldots, n)$,
Fitness function Fn, f_n, r_1, r_2, r_3, R, C_1, C_2, C_3, C_4, C_5, t=1, and max iterations *iter_max*
Evaluate fitness function Fn for each BL_i
Find best bird BL_{best}
While t < *iter_max* **do**
  for (i =1; $i \leq n$) **do**
    If (R < 0.5) **then**
      Update Location of the gey wolf agents using:
      $\vec{D}_\alpha = \left|\vec{D}_1 * \vec{F}_\alpha - \vec{F}\right|, \vec{D}_\beta = \left|\vec{D}_2 * \vec{F}_\beta - \vec{F}\right|, \vec{D}_\delta = \left|\vec{D}_3 * \vec{F}_\delta - \vec{F}\right|$
    else
      Update Speed of the flying bird using:
      $BS(t+1) = C_3 BS(t) + C_4 r_1 (BL_{best}(t) - BL_{nd}(t))$
      $+C_5 r_1 (BL_{Gbest} - BL_{nd}(t))$
      Update Location of the swimming bird using:
      $BL_{nd}(t+1) = r_1 + z * r_2 + (1 - z) * r_3 + BS(t+1)$
    end for

(Continued)

Algorithm 1: Continued

```
        end for
        Evaluate fitness function Fn for each BL_i
        Update R, r1, r2, r3, c, C1, C2
        Find best bird BL_best
        Set BL_Gbest = BL_best
        Set t = t + 1
    end while
    return BL_Gbest
```

Grey wolves may track down their victim and then completely engulf it. In most hunts, the alpha is in command of everything. During a hunting trip, the beta and delta occasionally join into the process. However, the location of the prey within the 2D search region is unknown. A statistical model of gray wolf hunting behavior will assume that all three wolves are aware of where to seek prey. Consequently, we will maintain the top three search results and force all of the grey wolves (including omegas) to recalculate their positions based on them. Here are the formulas we devised to do this:

$$\vec{D}_\alpha = \left|\vec{D}_1 * \vec{F}_\alpha - \vec{F}\right|, \vec{D}_\beta = \left|\vec{D}_2 * \vec{F}_\beta - \vec{F}\right|, \vec{D}_\delta = \left|\vec{D}_3 * \vec{F}_\delta - \vec{F}\right| \tag{9}$$

$$\vec{F}_1 = \vec{F}_\alpha - \vec{A}_1 * \vec{D}_\alpha, \vec{F}_2 = \vec{F}_\beta - \vec{A}_2 * \vec{D}_\beta, \vec{F}_3 = \vec{F}_\delta - \vec{A}_3 * \vec{D}_\delta \tag{10}$$

$$\vec{F}(t+1) = \frac{\vec{F}_1 + \vec{F}_2 + \vec{F}_3}{3} \tag{11}$$

A grey wolf (a search agent) is seen in Fig. 2 moving around in a 2D search space according to alpha, beta, and delta. Fig. 2 shows that the ultimate location of a grey wolf (search agent) will be in a random location inside the search area given by the coordinates of alpha, beta, and delta. Prey's location is estimated by alpha, beta, and delta wolves; other wolves follow this guess and update their locations around the prey at random.

3.5 Dataset Balancing Using SMOTE

The number of normal cases in the RPL-NIDDS17 dataset is far higher than the number of attack instances. So the routing assault is reduced to minority status while regular traffic gains control. Unbalanced data is the term used to describe this sort of information. A normal class dominates the classifier, which diminishes the chances of detecting intrusions. Classification will be skewed in favor of the typical class if the dataset is unbalanced. As a result, the intrusion class will have lower classification accuracy than the regular class. There are a number of ways to address this problem, including balancing the dataset. SMOTE, a modern algorithm, is widely used by numerous academics. To balance the dataset, it is possible to over-sample the minority classes by duplicating randomly selected k-nearest neighbors (KNNs). LSH-SMOTE identifies KNNs in a sample of data from an attack class. Nearest neighbors are combined with existing instances of attack class to create new examples of attack class. The original dataset is then supplemented with fake examples. Classification models are trained using the new, oversampled dataset. Tab. 1 provides a detailed information of the RPL-NIDDS17 dataset. Classification was based on 1,33,348 "normal" cases. Classification was based on 33,337 different attacks. Attack class has far less cases compared to the regular class. Thus, in the attack class, LSH-SMOTE based oversampling is used to ensure that the dataset is evenly distributed. $K = 4$ is used in the LSH-SMOTE method to increase the number of attack instances in order to match

the normal class's total number of instances. The number of assault occurrences rises to 1,33,348 after LSH-SMOTE-based oversampling.

Table 1: Dataset balancing using SMOTE

Category	Total instances in dataset	Utilized instances	Using proposed optimized LSH-SMOTE
Normal	4,31,981	1,33,348	1,33,348
Attack	33,337	33,337	1,33,348

4 Experimental Results

A Windows 11 laptop with an Intel Core i5 CPU clocked at 2.33 GHz, and 16 GB of RAM is used for the tests. In order to develop and evaluate the proposed framework, MATLAB R2020a was employed. MATLAB's Text Analytics Toolbox is utilized to do preprocessing on the dataset. This section evaluates and compares the performance of the proposed approach with other competing approaches.

NIDS performance with and without the LSH-SMOTE algorithm is shown in Tab. 2. With 98.1% accuracy, the proposed approach is far more accurate than the approach of traditional LSH-SMOTE and without LSH-SMOTE. There are also greater gains in terms of LSH-SMOTE algorithm accuracy, recall, F-score, specificity, and sensitivity than the other models. Using the proposed LSH-SMOTE, the performance of the NIDS is greatly enhanced.

Table 2: Evaluation of the results achieved with and without SMOTE

Metrics	Without LSH-SMOTE	With LSH- SMOTE	Proposed optimized LSH-SMOTE
Precision	0.6929	0.9526	0.978
Accuracy	0.8884	0.9640	0.981
F-score	0.7399	0.9645	0.986
Specificity	0.912	0.9514	0.978
Sensitivity	0.7939	0.9767	0.994

4.1 Evaluation Metrics

Based on four fundamental evaluation criteria, the performance of the proposed approach has been evaluated, namely: FN (false negative), FP (false positive), TN (true negative), and TP (true positive). These fundamental criteria are used to measure the five evaluation criteria, namely, accuracy, precision, sensitivity, specificity, and F-score.

- FN: refers to the number of incorrectly detected normal instances.
- FP: refers to the number of incorrectly detected attack instances.
- TN: refers to the number of correctly detected normal instances.
- TP: refers to the number of correctly detected attack instances.

Quantifying the model's capacity to accurately forecast all of the cases is referred to as accuracy. The measurement of this metric is applied as follows.

$$Accuracy = \frac{TN + TP}{FN + FP + TN + TP} \tag{12}$$

The number of accurately identified attack instances over the total attack instances in test data is called precision, and it is measured as follows.

$$Precision = \frac{TP}{FP + TP} \tag{13}$$

This metric is used to calculate the detection capabilities of the attack detection process. False positives are counted as less than true positives.

$$Sensitivity = \frac{TP}{FN + TP} \tag{14}$$

This metric refers to how many correctly categorized normal instances are there over the number of correctly classified normal instances.

$$Specificity = \frac{TN}{TN + FP} \tag{15}$$

The measurement of the F-score metric is calculated as follows

$$F - Score = 2 \times \frac{ADR \times Precision}{ADR + Precision} \tag{16}$$

4.2 Achieved Results

The proposed optimization algorithm is used to optimize the process of LSH-SMOTE for dataset balancing. The balanced dataset is used to train a KNN classifier to measure the effectiveness of the proposed approach. Tab. 3 presents the statistical analysis of the achieved results using the proposed approach and other approaches. As presented in this table, the results using the proposed approach outperform the other methods. The mean accuracy of the proposed approach is (98.1%), whereas the mean of the other approach is (96%).

Table 3: Statistical analysis of the achieved results using the proposed approach

	NO SMOTE	LSH-SMOTE	Proposed optimized LSH-SMOTE
Number of values	15	15	15
Minimum	0.8484	0.934	0.981
Median	0.8884	0.964	0.981
Maximum	0.8998	0.964	0.981
Mean	0.8838	0.96	0.981
Range	0.05144	0.03	0
Sum	13.26	14.4	14.72
10% Percentile	0.8544	0.94	0.981
25% Percentile	0.8884	0.964	0.981
75% Percentile	0.8884	0.964	0.981
90% Percentile	0.893	0.964	0.981

(Continued)

Table 3: Continued

	NO SMOTE	LSH-SMOTE	Proposed optimized LSH-SMOTE
Std. Deviation	0.01314	0.009103	0
Std. Error of mean	0.003393	0.00235	0
Geometric SD factor	1.015	1.01	1
Geometric mean	0.8837	0.96	0.981
Quadratic mean	0.8839	0.96	0.981
Harmonic mean	0.8836	0.9599	0.981
Kurtosis	3.842	4.664	
Skewness	−2.031	−2.308	
Coefficient of variation	1.487%	0.9482%	0.000%

On the other hand, the Wilcoxon signed-rank test is performed, and the results are presented in Tab. 4. As shown in this table, it can be noted that the proposed approach is statistically significant, the same as the other approaches. Therefore, the proposed approach can be used for the task of dataset balancing. In addition, the one way analysis of variance (ANOVA) test is performed, and the results are shown in Tab. 5. Based on the hypotheses of these tests, these tests emphasize the statistical effectiveness and significance of the proposed approach.

Table 4: Results of the Wilcoxon signed-rank test

	NO SMOTE	SMOTE	Proposed optimized LSH-SMOTE
Number of values	15	15	15
Actual median	0.8884	0.964	0.981
Theoretical median	0	0	0
Discrepancy	0.8884	0.964	0.981
Sum of negative ranks	0	0	0
Sum of positive ranks	120	120	120
Sum of signed ranks (W)	120	120	120
P value (two tailed)	<0.0001	<0.0001	<0.0001
Exact or estimate?	Exact	Exact	Exact
Significant (alpha = 0.05)?	Yes	Yes	Yes

Table 5: One-way analysis of variance test

ANOVA table	SS	DF	MS	F (DFn, DFd)	*P* value
Treatment (between columns)	0.07843	2	0.03921	F (2, 42) = 460.4	$P < 0.0001$
Residual (within columns)	0.003578	42	8.52E-05		
Total	0.082	44			

Moreover, to visualize the effectiveness and superiority of the proposed approach, Fig. 3 visualizes the achieved results. In this figure, the residual and homoscedasticity plots are used to visualize the accuracy of the proposed approach, as the residual error is minimal for the predicted results. Other plots such as QQ, heatmap, ROC, and accuracy range plots are used to show the effectiveness of the proposed approach. In these plots, the proposed method outperforms the other approaches.

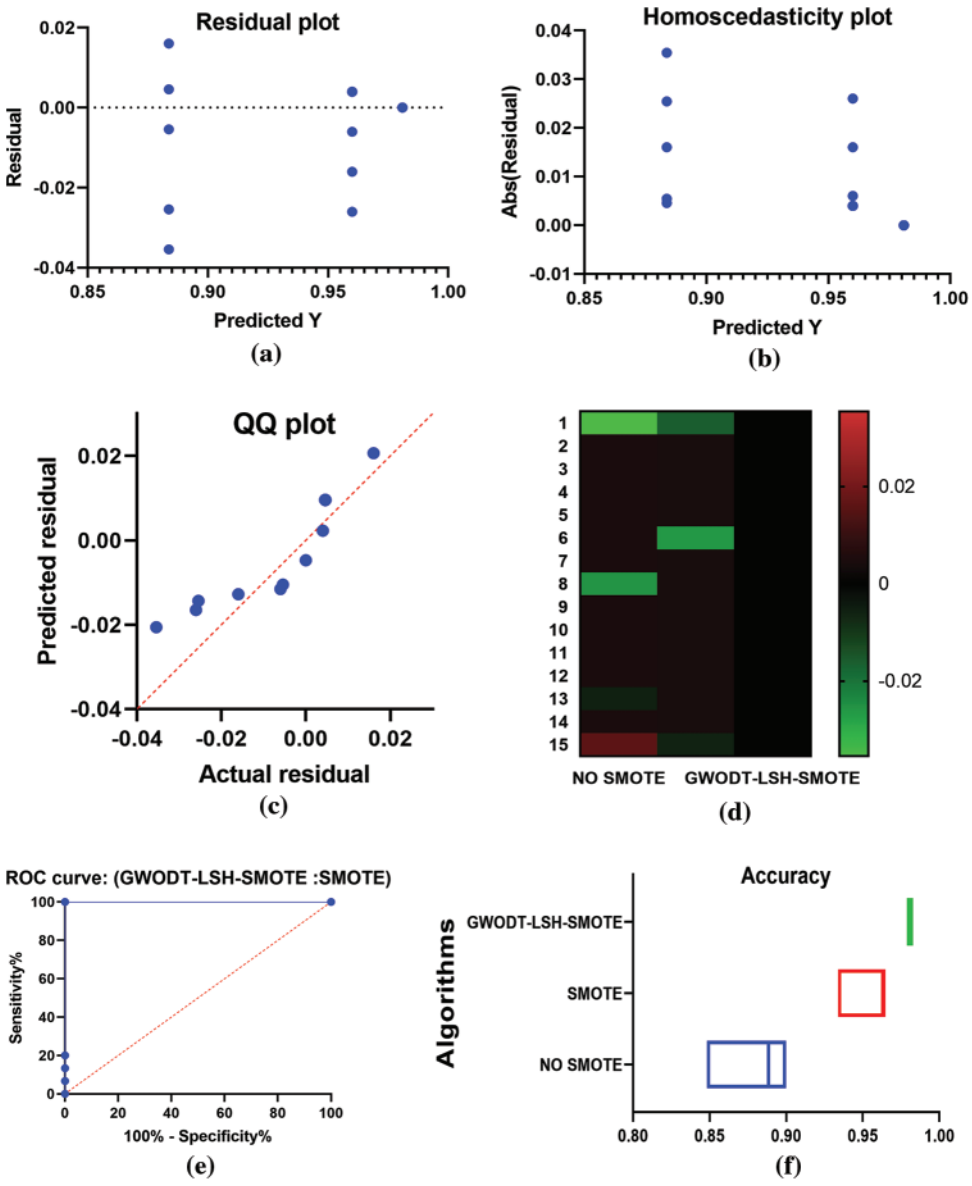

Figure 3: Visualization of the achieved results using the proposed approach

The histogram of the accuracy achieved by the proposed approach and other approaches is depicted in Fig. 4. In this figure, the proposed approach could achieve the highest accuracy, which is better than the other approaches. These results emphasize the effectiveness and superiority of the proposed approach.

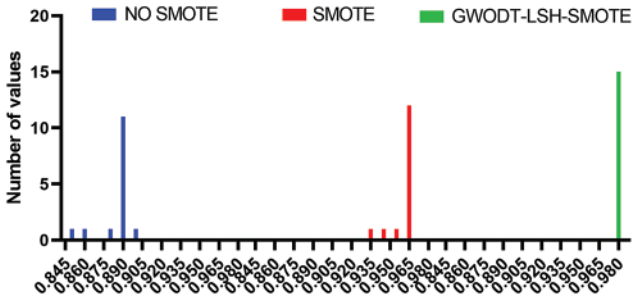

Figure 4: Histogram of the accuracy achieved by the proposed approach and other approaches

5 Conclusions

In this paper, we proposed a new approach for optimizing the LSH-SMOTE method to boost the performance of the data balancing step and improve the overall accuracy of classifying the attacks in network intrusion detection systems. All of the attacks included in this research include Sybil, blackhole, sinkhole, selective forwarding, local repair, and hello flooding. The proposed optimization method is based on grey wolf and dipper throated optimization. Using the RPL-NIDDS17 dataset, the proposed approach is trained and tested using various evaluation criteria such as accuracy, F-score, sensitivity, specificity, and precision. Experimental results show that the proposed approach is capable of detecting attacks on IoT networks based on RPL more accurately than the other approaches.

Acknowledgement: Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R323), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Funding Statement: Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R323), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] K. Ashton, "That 'internet of things' thing," *RFID Journal*, vol. 22, no. 7, pp. 97–114, 2009.
[2] L. D. Xu, W. He and S. Li, "Internet of things in industries: A survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2233–2243, 2014.
[3] J. Granjal, E. Monteiro and J. S. Silva, "Security for the internet of things: A survey of existing protocols and open research issues," *IEEE Communications Surveys Tutorials*, vol. 17, no. 3, pp. 1294–1312, 2015.
[4] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.
[5] A. Whitmore, A. Agarwal and L. Da Xu, "The internet of things-a survey of topics and trends," *Information Systems Frontiers*, vol. 17, no. 2, pp. 261–274, 2015.
[6] A. Verma and V. Ranga, "ELNIDS: Ensemble learning based network intrusion detection system for RPL based internet of things," in *Int. Conf. on Internet of Things: Smart Innovation and Usages (IoT-SIU)*, Ghaziabad, India, pp. 1–6, 2019.
[7] A. Verma and V. Ranga, "Security of RPL based 6LoWPAN networks in the internet of things: A review," *IEEE Sensors Journal*, vol. 20, no. 11, pp. 5666–5690, 2020.

[8] Z. Royaee, H. Mirvaziri and A. Bardsiri, "Designing a context-aware model for RPL load balancing of low power and lossy networks in the internet of things," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 2, pp. 2449–2468, 2021.
[9] A. Verma and V. Ranga, "Evaluation of network intrusion detection systems for RPL based 6LoWPAN networks in IoT," *Wireless Personal Communications*, vol. 108, no. 3, pp. 1571–1594, 2019.
[10] S. Cakir, S. Toklu and N. Yalcin, "RPL attack detection and prevention in the internet of things networks using a GRU based deep learning," *IEEE Access*, vol. 8, pp. 183678–183689, 2020.
[11] F. Yavuz, D. Ünal and E. Gül, "Deep learning for detection of routing attacks in the internet of things," *The International Journal of Computational Intelligence Systems*, vol. 12, no. 1, pp. 39–58, 2018.
[12] A. Verma and V. Ranga, "Machine learning based intrusion detection systems for IoT applications," *Wireless Personal Communications*, vol. 111, no. 4, pp. 2287–2310, 2020.
[13] B. Tama, M. Comuzzi and K. Rhee, "TSE-IDS: A two-stage classifier ensemble for intelligent anomaly-based intrusion detection system," *IEEE Access*, vol. 7, pp. 94497–94507, 2019.
[14] J. Yang, Y. Sheng and J. Wang, "A GBDT-paralleled quadratic ensemble learning for intrusion detection system," *IEEE Access*, vol. 8, pp. 175467–175482, 2020.
[15] B. Bhati, C. Rai, B. Balamurugan and F. Al-Turjman, "An intrusion detection scheme based on the ensemble of discriminant classifiers," *Computers & Electrical Engineering*, vol. 86, no. 1, pp. 1–9, 2020.
[16] E. -S. M. El-kenawy, A. Ibrahim, S. Mirjalili, M. Eid and S. Hussein, "Novel feature selection and voting classifier algorithms for COVID-19 classification in CT Images," *IEEE Access*, vol. 12, pp. 179317–179335, 2020.
[17] A. Davahli, M. Shamsi and G. Abaei, "Hybridizing genetic algorithm and grey wolf optimizer to advance an intelligent and lightweight intrusion detection system for IoT wireless networks," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 11, pp. 5581–5609, 2020.
[18] A. Al-Abassi, H. Karimipour, A. Dehghantanha and R. Parizi, "An ensemble deep learning-based cyber-attack detection in industrial control system," *IEEE Access*, vol. 49, pp. 83965–83973, 2020.
[19] A. Shahraki, M. Abbasi and O. Haugen, "Boosting algorithms for network intrusion detection: A comparative evaluation of real AdaBoost, gentle AdaBoost and modest AdaBoost," *Engineering Applications of Artificial Intelligence*, vol. 94, pp. 1–14, 2020.
[20] S. Kasongo and Y. Sun, "Performance analysis of intrusion detection systems using a feature selection method on the UNSW-NB15 dataset," *Journal of Big Data*, vol. 7, no. 1, pp. 1–20, 2020.
[21] Y. Zhou, G. Cheng, S. Jiang and M. Dai, "Building an efficient intrusion detection system based on feature selection and ensemble classifier," *Computer Networks*, vol. 174, no. 1, pp. 1–13, 2020.
[22] M. Asadi, M. Jamali, M. Parsa and V. Majidnezhad, "Detecting botnet by using particle swarm optimization algorithm based on voting system," *Future Generation Computer Systems*, vol. 107, no. 1, pp. 95–111, 2020.
[23] P. Kumar, G. Gupta and R. Tripathi, "An ensemble learning and fog-cloud architecture driven cyber-attack detection framework for IoMT networks," *Computer Communications*, vol. 166, no. 1, pp. 110–124, 2021.
[24] C. Pu, "Sybil attack in RPL-based internet of things: Analysis and defenses," *IEEE Internet of Things Journal*, vol. 7, no. 6, pp. 4937–4949, 2020.
[25] S. Murali and A. Jamalipour, "A lightweight intrusion detection for Sybil attack under mobile RPL in the internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 379–388, 2019.
[26] D. Gothawal and S. Nagaraj, "Anomaly-based intrusion detection system in RPL by applying stochastic and evolutionary game models over IoT environment," *Wireless Personal Communications*, vol. 110, no. 3, pp. 1323–1344, 2020.
[27] E. -S. M. El-Kenawy, S. Mirjalili, A. Ibrahim, M. Alrahmawy, M. El-Said *et al.,* "Advanced meta-heuristics, convolutional neural networks, and feature selectors for efficient COVID-19 X-ray chest image classification," *IEEE Access*, vol. 9, no. 1, pp. 36019–36037, 2021.
[28] A. Abdelhamid and S. Alotaibi, "Optimized two-level ensemble model for predicting the parameters of metamaterial antenna," *Computers, Materials & Continua*, vol. 73, no. 1, pp. 917–933, 2022.

[29] A. Abdelhamid and S. R. Alotaibi, "Robust prediction of the bandwidth of metamaterial antenna using deep learning," *Computers, Materials & Continua*, vol. 72, no. 2, pp. 2305–2321, 2022.
[30] D. Sami Khafaga, A. Ali Alhussan, E. M. El-kenawy, A. E. Takieldeen, T. M. Hassan *et al.,* "Meta-heuristics for feature selection and classification in diagnostic breast cancer," *Computers, Materials & Continua*, vol. 73, no. 1, pp. 749–765, 2022.
[31] D. Sami Khafaga, A. Ali Alhussan, E. M. El-kenawy, A. Ibrahim, S. H. Abd Elkhalik *et al.,* "Improved prediction of metamaterial antenna bandwidth using adaptive optimization of LSTM," *Computers, Materials & Continua*, vol. 73, no. 1, pp. 865–881, 2022.
[32] E. -S. M. El-Kenawy, S. Mirjalili, F. Alassery, Y. Zhang, M. Eid *et al.,* "Novel meta-heuristic algorithm for feature selection, unconstrained functions and engineering problems," *IEEE Access*, vol. 10, pp. 40536–40555, 2022.
[33] A. Abdelhamid, E. -S. M. El-kenawy, B. Alotaibi, M. Abdelkader, A. Ibrahim *et al.,* "Robust speech emotion recognition using CNN+LSTM based on stochastic fractal search optimization algorithm," *IEEE Access*, vol. 10, pp. 49265–49284, 2022.
[34] A. Verma and V. Ranga, "RPL-NIDDS17-A data set for intrusion detection in RPL based 6LoWPAN networks," *Internet of Things*, vol. 108, no. 1, pp. 1571–1594, 2018.
[35] H. Sun and R. Grishman, "Lexicalized dependency paths based supervised learning for relation extraction," *Computer Systems Science and Engineering*, vol. 43, no. 3, pp. 861–870, 2022.
[36] H. Sun and R. Grishman, "Employing lexicalized dependency paths for active learning of relation extraction," *Intelligent Automation & Soft Computing*, vol. 34, no. 3, pp. 1415–1423, 2022.
[37] S. S. M. Ghoneim, T. A. Farrag, A. A. Rashed, E. -S. M. El-Kenawy and A. Ibrahim, "Adaptive dynamic meta-heuristics for feature selection and classification in diagnostic accuracy of transformer faults," *IEEE Access*, vol. 9, pp. 78324–78340, 2021.
[38] H. Hassan, A. I. El-Desouky, A. Ibrahim, E. -S. M. El-Kenawy and R. Arnous, "Enhanced QoS-based model for trust assessment in cloud computing environment," *IEEE Access*, vol. 8, no. 1, pp. 43752–43763, 2020.
[39] M. M. Eid, E. -S. M. El-Kenawy and A. Ibrahim, "A binary sine cosine-modified whale optimization algorithm for feature selection," in *4th National Computing Colleges Conf. (NCCC 2021)*, Taif, Saudi Arabia, pp. 1–6, 2021.
[40] E. -S. M. El-Kenawy, S. Mirjalili, S. S. M. Ghoneim, M. M. Eid, M. El-Said *et al.,* "Advanced ensemble model for solar radiation forecasting using sine cosine algorithm and Newton's laws," *IEEE Access*, vol. 9, pp. 115750–115765, 2021.
[41] A. Salamai, E. -S. M. El-kenawy and A. Ibrahim, "Dynamic voting classifier for risk identification in supply chain 4.0," *CMC-Computers, Materials & Continua*, vol. 69, no. 3, pp. 3749–3766, 2021.
[42] A. Ibrahim, S. Mirjalili, M. El-Said, S. S. M. Ghoneim, M. Al-Harthi *et al.,* "Wind speed ensemble forecasting based on deep learning using adaptive dynamic optimization algorithm," *IEEE Access*, vol. 9, pp. 125787–125804, 2021.