A Novel Method for Anomaly Detection in the Internet of Things using Whale Optimization Algorithm

Zhihui Zhu[*1], Meifang Zhu[2]
Guangzhou Maritime University, Guangzhou 510725, Guangdong, China[*1]
Guangdong Lingnan Institute of Technology, Guangzhou 511510, Guangdong, China[2]

***Abstract*—The Internet of Things (IoT) is integral to human life due to its pervasive applications in home appliances, surveillance, and environment monitoring. Resource-constrained IoT devices are easily accessible to attackers due to their direct connection to the unsafe Internet. Public access to the Internet makes IoT objects more susceptible to intrusion. As the name implies, anomaly detection systems are designed to identify anomalous traffic patterns that conventional firewalls fail to detect. Effective Intrusion Detection Systems (IDSs) design faces three major problems, including handling high dimensionality, selecting a learning algorithm, and comparing entered observations and traffic patterns using a distance or similarity measure. Considering the dynamic nature of the entities involved and the limited computing resources available, more than traditional anomaly detection approaches is required. This paper proposes a novel method based on Whale Optimization Algorithm (WOA) to detect anomalies in IoT-based networks that conventional firewall systems cannot detect. Experiments are conducted on the KDD dataset. The accuracy of the proposed method is compared for classifiers such as kNN, SVM, and DT approaches. The detection accuracy rate of the proposed method is significantly higher than that of other methods for DoS, probing, normal attacks, R2L attacks, and U2R attacks compared to other methods. This method shows an impressive increase in accuracy when detecting a wide range of malicious activities, from DoS, probing, and privilege escalation attacks, to remote-to-local and user-to-root attacks.**

Keywords—Internet of things; anomaly detection; intrusion detection; firewall; whale optimization algorithm; accuracy

I. Introduction

Scientific and technological advancements in the fields of optical networks [1, 2], Internet of Things (IoT) [3], cloud computing, Complementary Metal-Oxide Semiconductor (CMOS) [4, 5], machine learning [6], 5G connectivity [7, 8], Blockchain [9], artificial intelligence [10, 11], and smart grids [12] have greatly benefited society. In recent years, the internet has grown tremendously and is now used to connect objects. The Internet of Things (IoT) influences almost every aspect of human and industrial life [13, 14]. By linking physical things together, the IoT is expected to bridge various technologies [15]. Wireless technology advancements such as radio frequency identification (RFID), Bluetooth, and WiFi enable better communication among objects and with the internet [16, 17]. A unique identifier can also be assigned to each item [18]. A lack of security mechanisms and the Internet connectivity of IoT devices makes them vulnerable to attacks [19]. By gaining control over smart devices, an attacker can hack IoT devices and use them maliciously to hack other IoT devices [20]. As part of anomaly detection, intrusion detection examines incoming traffic for abnormality or abnormality [21]. In order to recognize abnormal traffic within a network efficiently, intrusion detection systems need to automate their detection procedures [22]. The majority of network intrusion detection systems analyze incoming traffic using data mining and clustering techniques. A fundamental function of an intrusion detection system is to identify normal or abnormal traffic patterns based on the current traffic pattern [23].

An Intrusion Detection System (IDS) tracks network activity in real-time and alerts or takes proactive action when suspicious transmissions are detected [24]. The main difference between IDS and other network security tools is that IDS can detect ongoing invasions as well as recent intrusions. An intrusion detection system generally distinguishes between normal and anomalous network traffic behavior and determines the type of attack based on a binary classification problem [25]. It is primarily motivated by improving classification accuracy by detecting intrusive behavior. Network information security has gradually gained attention over the past 30 years [26]. IDS systems are currently categorized as anomaly-based detection systems and signature-based detection systems. The signature-based detection system compares the signatures extracted from the subsequent detection systems with those extracted from known attack methods to detect upcoming attacks and notify users. While anomaly-based detection systems are accurate, they are limited in their ability to detect unidentified attacks, such as 0-DAY vulnerabilities and advanced persistent threats.

A classifier based on ensembles is proposed as a method for improving the accuracy of IDS. Twelve experts are trained and tested to form an ensemble. WOA weighs each expert's opinion. As a meta-optimizer, the LUS method finds high-quality parameters based on the behavioral parameters inserted by the user. The weights of each expert are then adjusted using WOA. Seven stages are involved in the system framework: data preprocessing, SVM classification, k-NN classification, decision tree classification, weighting with WOA, and comparison of results. The remainder of the paper is organized as follows. The Section II reviews related work. A detailed description of the proposed method appears in Section III. Section IV reports the results of the experiments. The paper is concluded in Section V.

II. Related Works

This section will review the existing anomaly and intrusion detection methods and determine their main features and weaknesses.

Alamiedy, et al. [28] have proposed an IDS scheme based on the Grey Wolf Optimization (GWO) algorithm. The GWO algorithm is employed for feature selection in order to identify the optimum dataset features for accurate classification. Besides, the support vector machine has been utilized in evaluating the accuracy of selected features in attack prediction. Experiments confirm that the offered method has obtained classification accuracy of 94%, 92%, 58%, and 54% for DoS, probing, R2L, and U2L attacks, respectively.

An IDS approach based on a genetic algorithm and Deep Belief Network (DBN) has been presented by Zhang, et al. [29]. When faced with multiple iterations of the genetic algorithm and varying attacks, generating several neurons in each layer and an optimal number of hidden layers, the proposed mechanism uses DBN to achieve a high detection rate while maintaining a compact structure. The performance of the method has been assessed based on the NSL-KDD dataset. The results indicate that the combined IDS and DBN model effectively reduced neural network complexity and improved intrusion detection rates.

Moreover, a random neural network-based IDS for IoT has been developed by Qureshi, et al. [30], in which, with the NSL-KDD dataset, neurons are trained and then tested at different rates of learning. The accuracy of RNN-IDS was increased from 86% to 96% by using two methods to evaluate the proposed approach. Simulation outcomes indicate that the proposed IDS can distinguish anomalous traffic more accurately from normal traffic.

An EFSAGOA method, which combines an Ensemble of Feature Selection (EFS) and Adaptive Grasshopper Optimization algorithm (AGOA), has been introduced by Dwivedi, et al. [31]. At first, in order to determine the highest-ranked attributes, the EFS method was applied to rank attributes. Using the AGOA method, key attributes derived from the reduced datasets were identified for network traffic prediction. To optimize the classification process, AGOA applies Support Vector Machines (SVM) as a fitness function. Additionally, the method was used to optimize the tube size, kernel parameter, and penalty factor of SVM classifiers. Utilizing ISCX 2012 dataset, the performance of EFSAGOA has been evaluated. In comparison to existing methods in ISCX 2012 data, the proposed method produced better accuracy, false alarm rates, and detection rates.

A novel host-based automated framework for IDS in the IoT has been presented by Gassais, et al. [32], in which user and kernel space information are combined with machine learning approaches to identify intrusions of different types. Tracing methods have been utilized to detect the behavior of devices automatically, transform data into numeric arrays, and train machine learning algorithms. Several machine learning algorithms have been implemented to improve detection capability with minimal overhead on monitored devices.

Furthermore, a novel IDS combining deep learning and a dendritic cell algorithm has been proposed by Aldhaheri, et al. [33]. Classifying IoT intrusions and preventing false alarms are the main aims of the approach. By selecting the appropriate set of features from the IoT-Bot dataset, the proposed IDS categorizes signals and then performs classification using the dendritic cell algorithm. The proposed approach demonstrated a better ability to detect IoT attacks, achieving an accuracy rate of 97% and a low false positive rate.

Brown and Anwar [34] have developed a deep neural network-based validation model integrated into an artificial immune system based on human intelligence. The solution provides implementation strategies and a pilot implementation of the core component to address the challenges associated with IoT networks. The suggested approach is suitable for discovering real-time attacks and is adaptable to changing network environments. This mechanism may serve as a baseline for the development of holistic IoT IDS in which each node plays a role in network security.

Finally, Ge, et al. [24] have offered a novel IDS for IoT utilizing a customized deep-learning technique. They have utilized an innovative IoT dataset of real-world attacks, such as data theft and denial of service attacks. They have developed a feed-forward neural network model embedded with multi-class classification layers. Besides, a binary classifier based on a second feed-forward neural network model was built using transfer learning to encode categorical features of high dimensions. For both binary and multi-class classifiers, the proposed method achieves higher classification accuracy.

III. Proposed Method

In this section, at first, the problem statement is described as well as the adopted network model is explained. Then, the suggested strategy is clarified step by step.

A. Problem Definition

With the rise of IoT applications and smart objects, IoT networks generate more data and traffic, resulting in a rise in IoT vulnerabilities and, consequently, RPL threats. Although RPL offers mechanisms for achieving confidentiality, integrity, and replay protection through encryption of control messages, local and global repairs, and loop detection, it is still susceptible to internal attacks. The RPL network has vulnerabilities beyond its encryption and authentication defenses. The second line of defense for networks is IDSs, which monitor network activity and node behavior for disruption attempts.

B. Network Model

First-line defenses against computer system cyberattacks are security frameworks that enforce industry standards such as authentication, authorization, and confidentiality. Vulnerabilities in system software, operational errors, and other issues may make attacks more likely. IDSs are critical in identifying and alerting system administrators to such attacks. Depending on the configuration, an IDS can be installed on individual hosts, at a central location, or distributed throughout the network. Fig. 1 illustrates how IDS operates in several areas across the network system. The IDS is a kind of IDS intended to detect attacks on a computer network rather than a single system. It is designed to detect malicious activities such as unauthorized access, data manipulation, and denial of service attacks. It monitors the network for suspicious activities and flags any potential threats, allowing for quick response and mitigation of possible damage. These systems monitor network operations using network telemetry, which may comprise network traffic, network flow metadata, and host event logs to identify attack events. By analyzing this telemetry, the system can detect and classify malicious activity, alerting administrators to potential malicious activity and allowing for remediation of any potential threats [35].

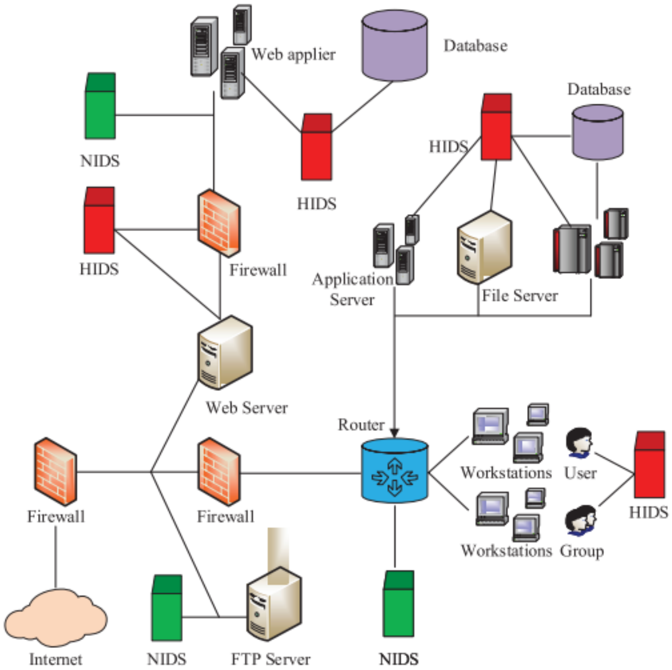


Fig. 1. Intrusion detection system and different places of the network [35].

An IDS funnels all network traffic via its sensors to identify intrusions and anomalies. As network traffic increases, using a single IDS on a network poses congestion issues if the network throughput is too high. Deep Packet Inspection may include significant pattern matching against complicated attack rule signatures. *Pattern matching* is a time-consuming procedure that requires substantially more computer resources than a firewall, which might cause an IDS to become overloaded. When an IDS becomes overburdened and begins dropping or ignoring packet content, it might compromise the network's security. Eventually, some intrusions may go unnoticed since some packets associated with the same attack may evade the IDS's inspection, leading to incomplete packet matching [6]. There are several strategies for handling high levels of network traffic for IDS, including:

C. Proposed Algorithm Description

As stated earlier, this paper aims to improve the accuracy of IDS by developing ensemble-based classifiers. An ensemble of twelve experts is formed after training and testing twelve experts. Each expert's opinion is weighed according to WOA. User-inputted behavioral parameters are a vital indicator of the effectiveness of WOA. Each expert's weights are then adjusted based on the improved WOA. A seven-stage system framework is designed to simplify the process: preprocessing data, classifying data with five distinct SVM experts, classifying data with five distinct KNN experts, classifying data with five distinct decision tree classifiers, setting weights with WOA, and comparing the results.

1) Adopted dataset: Experiments are conducted using the Knowledge Discovery and Data Mining 1999 (KDD99) dataset. Thousands of records describe connections in the dataset. Each TCP/IP connection contains 41 qualitative and quantitative features. Observations are classified as normal or intrusive based on their features. The performance of each classifier must be evaluated on two datasets: training and testing. The data is taken from [36]. The KDD99 dataset contains four types of attacks.

- User to Root (U2R): connects an attacker to the root account after gaining access.
- Remote to Local (R2L): An entry attempt into a computer or network illegally.
- Probe (Probing): Examining the target machine for potential weaknesses.
- Denial of Service (DoS): The attempt to deny authorized users access to a targeted computer's services.

2) Data preprocessing: Each observation must have a set of numerical values in order to be classified using the proposed methods. Additionally, each class must be given a numerical value. The proposed classification algorithms are incompatible with three symbolic features of KDD99 data:

- Flag: The connection status flag is represented by this feature.
- Service: It represents a destination service (for example, telnet, FTP, etc.).
- Protocol type: This feature signifies the connection protocol.

Data preprocessing involves two steps: data mapping and state identification.

- State identification: The KDD99 defines states for different features, such as regular connections or attacks. The data has five major classes: R2L, U2R, Probe, DoS, and Normal. Numerical values are assigned to each state.
- Data mapping: Every observation is mapped to a numerical value in the training, validation, and testing datasets. The three features are each given a numerical value between 1 and n, where n represents the symbol count.

3) SVM classifier: Support vector machines (SVM) can effectively solve classification and regression problems. This technique has a low generalization error and does not overfit training data. When a model performs poorly outside the training set, it is referred to as over-fitting or having a high generalization error. SVM is most effective when separating data sets linearly, which means instances in one class are all positioned along the same hyperplane H. SVM chooses the hyperplane H with the shortest distance between every pair of instances in each class. Up to this point, only linearly separable data have been considered. Such a hyperplane may only be possible for some real-life data sets. Such separation can be achieved using SVM based on data mapping to another feature space. In most cases, this transformation involves mapping into high-dimensional spaces. Kernel functions perform these modifications.

A multi-class SVM is extended by training five binary classifiers, one for each class. Suppose i = (1, . . ., 5) belongs to the quintuple F = (R2L, U2R, DoS, Probe, and Normal), and Bi denotes the binary classifier for target class *i* within *F*. Binary classifiers are trained on the entire training set for their respective target classes. Training the classifier Bi involves labeling observations belonging to class *i* as 1 and all other observations as 0. Classifying observations into one of the five classes is referred to as the One-versus-All approach. The 5-classifier set is used to distinguish between binary classifiers.t. According to Fig. 2, binary classifiers and experts have different relationships illustrated in their output formats. Binary classifiers take input data and output one of two possible classes, while experts take input data and output a continuous value. This difference in output formats reflects the different ways in which the two types of models process data.

In SVM, the RBF kernel function yields the best results [37]. Various RBF functions are employed in experiments to determine the performance of SVM classifiers with RBF kernel functions. In order to maximize the efficiency of the SVM algorithm, six different SVM experts are trained with different RBF parameters. In addition, this approach ensures that ensemble classifiers have a greater variety of experts. The RBF vector defines the selected values for RBF parameters as follows: RBF = [5, 2, 1, 0.5, 0.2, 0.1]. RBF vector values determine the accuracy of binary classifiers in each expert system. Six SVM experts are developed based on the RBF vector:

- SVM 1: RBF = 5
- SVM 2: RBF = 2
- SVM 3: RBF = 1
- SVM 4: RBF = 0.5
- SVM 5: RBF = 0.2
- SVM 6: RBF = 0.1

4) kNN classifier: An effective and simple tool for object classification is the k-nearest neighbor (kNN) algorithm [38]. Consider observations and targets (o_1, t_1), . . ., (o_n, t_n), where observations $o_i \in$ Rd and targets $t_i \in \{0, 1\}$. For a given i in the training sample, kNN predicts the test vector class based on the class labels of the nearest neighbors. A kNN classifies new points by identifying the points with the most votes based on the K closest points. The Euclidean distance is a distance metric commonly used in kNN to compare two vectors (points):

$$d^2(x_i, x_j) = \|x_i - x_j\|^2 = \sum_{k=1}^{d} (x_{ik} - x_{jk})^2 \quad (1)$$

In contrast to SVM, kNN classifiers can solve multi-class problems. However, five binary classifiers are needed to make kNN and SVM experts compatible. Accordingly, kNN expert systems are structured similarly to the SVM expert systems, as shown in Fig. 2. The compatibility of SVM and kNN expertise allows them to be combined into an ensemble expert system. kNN classifiers use the k parameter to determine how many neighbors close to a given observation are in a training set. The accuracy of binary classifiers inside an expert will vary as this parameter is changed. The kNN classifier can be optimized by creating six experts with different values of the *k* parameter as defined by the k vector: K = [1, 3, 5, 7, 9, 11]. The six k-NN experts are created as follows by selecting different k parameters:

- k-NN 1: k=1;
- k-NN 2: k=3;
- k-NN 3: k=5;
- k-NN 4: k=7;
- k-NN 5: k=9;
- k-NN 6: k=11;

5) Whale optimization algorithm for IDS: The WOA algorithm is a swarm-based intelligent algorithm for continuous optimization problems. Compared to recent meta-heuristics methods, it exhibits superior performance. It is more straightforward and robust than other swarm intelligence algorithms, making it comparable to other nature-inspired algorithms. A single parameter (time interval) is required to achieve the desired result in practice. WOA involves humpback whales searching for food in a multidimensional space. Humpback whale locations are considered decision variables, while distances between them and food are represented as objective costs. Three operational processes determine a whale's time-dependent location: search for prey, bubble-net attacking method, and shrinking encircling prey [39]. The primary presentation of the WOA is shown in Fig. 3. These operational processes are described and mathematically expressed in the following. A spiral mathematical formulation can describe the bubble-net behavior of humpback whales as follows.

$$\vec{X}(t+1) = \overrightarrow{D'}.e^{bl}.cos(2\pi l) + \overrightarrow{X^*}(t) \quad (2)$$

$$\vec{X}(t+1) = \begin{cases} \overrightarrow{X^*}(t) - \vec{A}.\vec{D} & if p < 0 \\ \overrightarrow{D'}.e^{bl}.cos(2\pi l) + \overrightarrow{X^*}(t) & if\ p \geq 0 \end{cases} \quad (3)$$

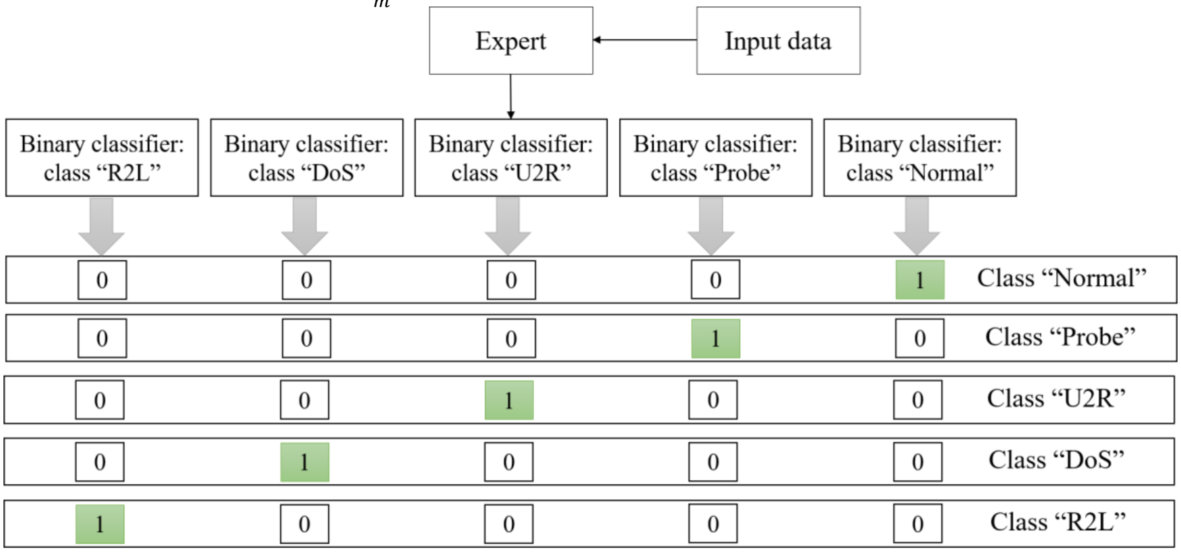
In Eq. (3), p is a constant used to explain the logarithmic spiral's shape, and k is a uniformly distributed number. As a global optimizer, if A > 1 or A < -1, a randomly chosen search agent replaces the best search agent as follows:

$$\vec{D} = \left|\vec{C}.\overrightarrow{X_{rand}} - \vec{X}\right| \tag{4}$$

$$\vec{X}(t+1) = \overrightarrow{X_{rand}} - \vec{X}.\vec{D} \tag{5}$$

The current iteration nominates $\overrightarrow{X_{rand}}$ arbitrarily from whales. Whales with a minimum fitness function represent the ideal solution. A whale with the best fitness represents the optimal set of weight coefficients w = (w_1, w_2,..., w_n), where n denotes the number of experts. This means that each whale has its own set of weight coefficients. Using Eq. (2), every observation *x* in the sample is classified according to the voting algorithm *y*. Every observation in the training set is provided with the correct class (target). As a training sample of size m is classified correctly, c is the number of instances where an output is predicted to have the same value as a target T, or y = T. Based on the validation sample, ACC(w) is the fraction of correctly classified observations, ACC (w) $=\frac{c}{m}$, where m is the number of observations. The accuracy of ensemble classifiers should be maximized, or the error minimized for each whale in order to achieve improved performance.

Weights are generated separately for each class. According to Fig. 2, the ensemble classifier created by WOA weights will have the same structure as an expert. Consequently, five weights need to be generated with WOA, one for each binary classifier in the base expert. Weights are generated based on the validation data. An accurate evaluation of the accuracy of classifiers based on training data is required. It is not acceptable to evaluate model performance using the same data for training since this would lead to strongly biased weights and could easily result in an overfitted model. The fitness value of an expert system cannot also be determined by testing data since it is necessary to use only testing targets to evaluate the performance of each expert system, whether it is a basic classifier or an ensemble. The validation dataset was created by taking a subset from the corrected.gz file used for testing and removing it from all testing datasets. This ensures the independence of the validation process.


Expert
Input data
Binary classifier: class "R2L"
Binary classifier: class "DoS"
Binary classifier: class "U2R"
Binary classifier: class "Probe"
Binary classifier: class "Normal"

0	0	0	0	1	Class "Normal"
0	0	0	1	0	Class "Probe"
0	0	1	0	0	Class "U2R"
0	1	0	0	0	Class "DoS"
1	0	0	0	0	Class "R2L"

Fig. 2. Expert system structure.

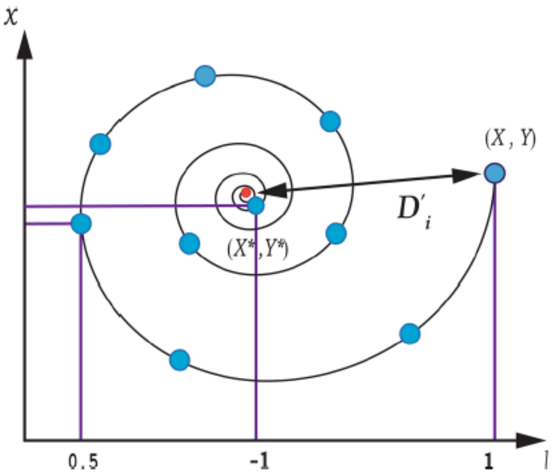


Fig. 3. Position update in a spiral.

IV. Experimental Results

This study was conducted using Matlab-2018 32bit on Windows 7 Professional 32bit, with an Intel Core i5 processor and 8GB of RAM. As mentioned earlier, the KDD-99 dataset is used in the proposed IDS strategy. KDD-99 consists of a large number of records that fall into five different categories. In fact, KDD-99 records fall into one of these five classes. The number of randomly selected records from the KDD-99 dataset is listed in Table I.

It should be noted that the selected dataset is divided into two groups of training and testing datasets. The training dataset is used to train the classifier, while the testing dataset is used to evaluate it. These datasets are shown in Table II and Table III.

A. Detection Accuracy Evaluation

The classification accuracy criterion is one of the main and most significant evaluation criteria for each IDS mechanism. Considering that the dataset considered in this article includes five different classes. As a result, each of the fifteen presented classifiers includes five binary classifiers. According to Eq. (6), it is possible to calculate the classification accuracy of each binary classifier. In this regard, A indicates the classification accuracy of the binary classifier, S indicates the total number of test samples of the desired class, and C indicates the number of correctly identified samples of the same class. The fifteen classifiers will be examined in the following according to their accuracy of identification.

$$A = \frac{C}{S} \tag{6}$$

- Classifiers based on support vector machine: Five different classifiers can be formed based on the SVM. These classifiers include a multi-class SVM based on the RBF kernel function with y values set to 0.1, 0.2, 0.5, 1, and 2. Table IV shows the accuracy of different SVM-based classifiers.
- Classifiers based on k-nearest neighbor: There are five types of kNN-based classifiers. Classifiers include multi-class kNNs with k values of 1, 3, 5, 7, and 9. The accuracy of these classifiers is shown in Table V.
- Classifiers based on decision tree: There are five types of classifiers based on the C4.5 classification algorithm. These classifiers contain 19, 21, 23, 25, and 27 features. Previous research has determined the number of selected features. In fact, the difference between these five classifiers is the number of selected features. Table VI shows the accuracy of these classifiers.
- Proposed algorithm: The proposed IDS strategy in this paper comprises fifteen different classifiers. By combining these classifiers, the proposed IDS system is designed and built. Each classifier is also given a suitable weight based on the WOA. The proposed system based on the WOA was trained with 70% of the data and tested with 30% of the data. Table VII shows the number of training data, the test results, and the recognition accuracy. This proposed method outperforms fifteen different classifiers regarding average detection accuracy, as shown in Fig. 4.

B. Performance Analysis on the UNSW-NB15 Dataset

This scenario tests the efficiency of the ensemble-based WOA model in terms of accuracy, F-measure, precision, recall, and AUC. The UNSW-NB15 dataset contains 175,341 records for training and 82,332 records for testing. Similarly, to the NSL-KDD dataset, the ensemble classifier with WOA demonstrated superior performance for intrusion detection with an AUC of 99.6%, F-measure of 99%, recall of 99.1%, precision of 99.2%, and accuracy of 99.3%. Fig. 5 compares the ensemble-based WOA model with other approaches.

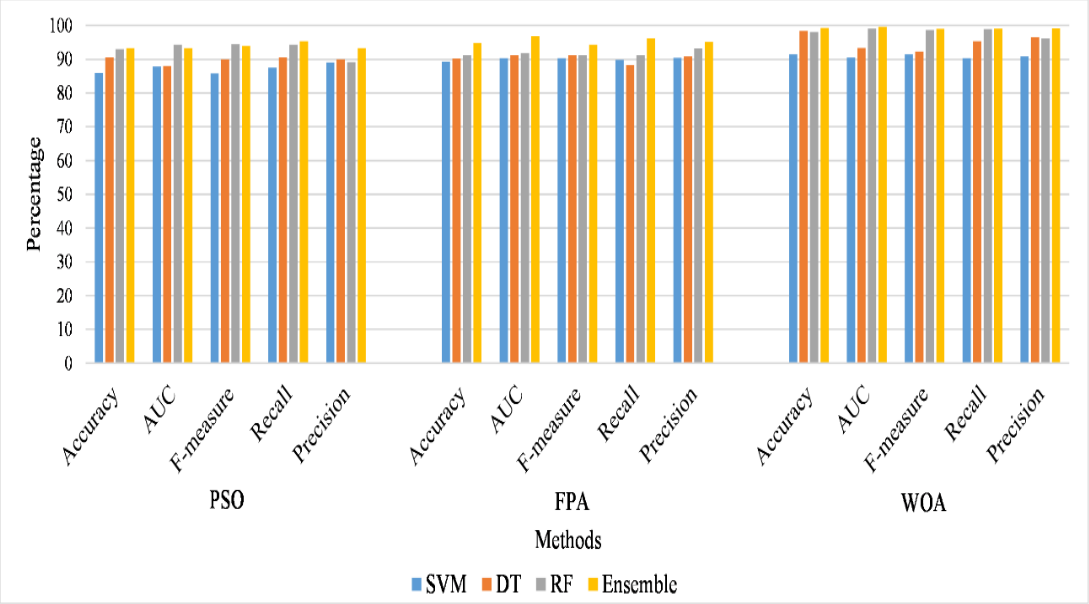


Fig. 4. Comparison results.

TABLE II. THE NUMBER OF RECORDS RANDOMLY SELECTED FROM KDD-99

Class	NC	DoS	R2L	U2R	Probing
Number of records	12500	7231	4876	104	12500

TABLE III. TRAINING DATASET

Class	NC	DoS	R2L	U2R	Probing
Number of records	5000	4107	1126	52	5000

TABLE IV. TESTING DATASET

Class	NC	DoS	R2L	U2R	Probing
Number of records	7500	3124	3750	52	7500

TABLE V. DETECTION ACCURACY OF SVM-BASED CLASSIFIERS

Classifier	NC	DoS	R2L	U2R	Probing
Classifier 1	68.55 %	93.26 %	81.44 %	99.88 %	92.1 %
Classifier 2	73.44 %	94.66 %	81.63 %	99.74 %	93.37 %
Classifier 3	76.69 %	98.88 %	81.43 %	99.45 %	94.31 %
Classifier 4	82.16 %	98.17 %	81.8 %	99.55 %	94.58 %
Classifier 5	76.55 %	94.55 %	81.18 %	99.18 %	94.69 %

TABLE VI. DETECTION ACCURACY OF KNN-BASED CLASSIFIERS

Classifier	NC	DoS	R2L	U2R	Probing
Classifier 6	81.74 %	97.8 %	83.93 %	99.65 %	96.2 %
Classifier 7	81.28 %	97.36 %	83.45 %	99.73 %	95.29 %
Classifier 8	76.44 %	93.54 %	83.44 %	99.77 %	92.3 %
Classifier 9	76.16%	92.18 %	83.55 %	99.78 %	92.32 %
Classifier 10	76.1 %	92.44 %	83.56 %	99.80 %	92.33 %

TABLE VII. DETECTION ACCURACY OF CLASSIFIERS BASED DECISION TREE

Classifier	NC	DoS	R2L	U2R	Probing
Classifier 11	78.37 %	90.45 %	81.65 %	99.1 %	92.5 %
Classifier 12	79.41 %	91.04 %	81.9 %	99.21 %	93.48 %
Classifier 13	80.55 %	91.5 %	82.48 %	99.43 %	94.51 %
Classifier 14	80.6 %	94.31 %	82.13 %	99.47 %	95.88 %
Classifier 15	81.73 %	95.77 %	83.82 %	99.68 %	95.31 %

TABLE VIII. THE NUMBER OF TRAINING AND TESTING DATA AND THE DETECTION ACCURACY OF THE PROPOSED METHOD

	NC	DoS	R2L	U2R	Probing
The number of records in the training dataset	8750	5062	3413	73	8750
The number of records in the testing dataset	3750	2169	1463	31	3750
Accuracy	90.2 %	98.66 %	89.61	99.9 %	96.83 %

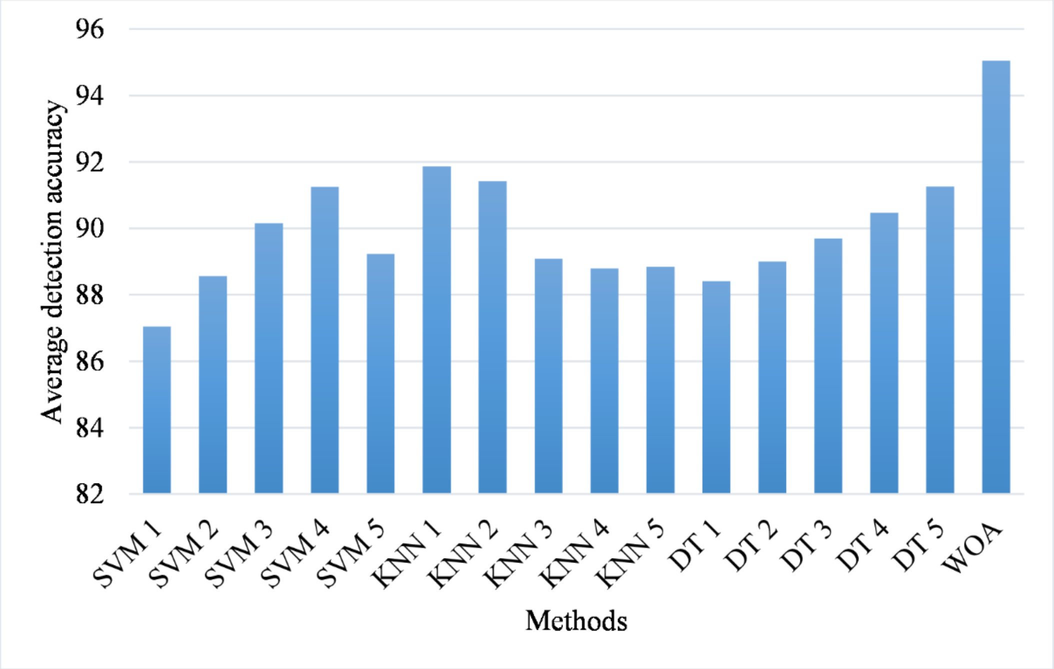


Fig. 5. Detection accuracy comparison.

V. Conclusion

The IoT enables physical objects in different domains to become Internet hosts, raising high expectations. Nevertheless, attackers may also use the IoT to threaten the privacy and security of users. Hence, the IoT requires security solutions. IDSs play a critical role in keeping IoT networks accessible and secure. This paper proposed a new strategy to improve the accuracy of IDS by developing ensemble-based classifiers. Twelve experts are trained and tested to form an ensemble. With LUS, user-supplied behavioral parameters are used as meta-optimizers to estimate high-quality parameters. WOA is then used to adjust the weights of each expert. The detection accuracy rates of the proposed method were significantly higher than those of other approaches for attacks, such as DoS, probing, normal, R2L, and U2R. We will investigate the efficiency of the ensemble-based WOA model using other intrusion datasets in the future, and apply this approach to other optimization problems as well.

References

[1] F. Khosravi, M. Tarhani, S. Kurle, and M. Shadaram, "Implementation of an Elastic Reconfigurable Optical Add/Drop Multiplexer based on Subcarriers for Application in Optical Multichannel Networks," in 2022 International Conference on Electronics, Information, and Communication (ICEIC), 2022: IEEE, pp. 1-4.

[2] F. Khosravi, G. Mahdiraji, M. Mokhtar, A. Abas, and M. Mahdi, "Improving the performance of three level code division multiplexing using the optimization of signal level spacing," Optik, vol. 125, no. 18, pp. 5037-5040, 2014.

[3] F. Kamalov, B. Pourghebleh, M. Gheisari, Y. Liu, and S. Moussa, "Internet of Medical Things Privacy and Security: Challenges, Solutions, and Future Trends from a New Perspective," Sustainability, vol. 15, no. 4, p. 3317, 2023.

[4] S. Seyedi and B. Pourghebleh, "A new design for 4-bit RCA using Quantum Cellular Automata Technology," Optical and Quantum Electronics, vol. 55, no. 1, p. 11, 2023.

[5] S. Seyedi, B. Pourghebleh, and N. Jafari Navimipour, "A new coplanar design of a 4-bit ripple carry adder based on quantum-dot cellular automata technology," IET Circuits, Devices & Systems, vol. 16, no. 1, pp. 64-70, 2022.

[6] J. Akhavan and S. Manoochehri, "Sensory data fusion using machine learning methods for in-situ defect registration in additive manufacturing: a review," in 2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), 2022: IEEE, pp. 1-10.

[7] P. He, N. Almasifar, A. Mehbodniya, D. Javaheri, and J. L. Webber, "Towards green smart cities using Internet of Things and optimization algorithms: A systematic and bibliometric review," Sustainable Computing: Informatics and Systems, vol. 36, p. 100822, 2022.

[8] I. Ataie, T. Taami, S. Azizi, M. Mainuddin, and D. Schwartz, "D 2 FO: Distributed Dynamic Offloading Mechanism for Time-Sensitive Tasks in Fog-Cloud IoT-based Systems," in 2022 IEEE International Performance, Computing, and Communications Conference (IPCCC), 2022: IEEE, pp. 360-366.

[9] S. Meisami, M. Beheshti-Atashgah, and M. R. Aref, "Using Blockchain to Achieve Decentralized Privacy In IoT Healthcare," arXiv preprint arXiv:2109.14812, 2021.

[10] F. Vahedifard, S. Hassani, A. Afrasiabi, and A. M. Esfe, "Artificial intelligence for radiomics; diagnostic biomarkers for neuro-oncology," World Journal of Advanced Research and Reviews, vol. 14, no. 3, pp. 304-310, 2022.

[11] S. A. Saeidi, F. Fallah, S. Barmaki, and H. Farbeh, "A novel neuromorphic processors realization of spiking deep reinforcement learning for portfolio management," in 2022 Design, Automation & Test in Europe Conference & Exhibition (DATE), 2022: IEEE, pp. 68-71.

[12] S. H. Haghshenas, M. A. Hasnat, and M. Naeini, "A Temporal Graph Neural Network for Cyber Attack Detection and Localization in Smart Grids," arXiv preprint arXiv:2212.03390, 2022.

[13] B. Pourghebleh, N. Hekmati, Z. Davoudnia, and M. Sadeghi, "A roadmap towards energy-efficient data fusion methods in the Internet of Things," Concurrency and Computation: Practice and Experience, p. e6959, 2022.

[14] T. Taami, S. Azizi, and R. Yarinezhad, "An efficient route selection mechanism based on network topology in battery-powered internet of things networks," Peer-to-Peer Networking and Applications, pp. 1-16, 2022.

[15] T. Taami, S. Krug, and M. O’Nils, "Experimental characterization of latency in distributed iot systems with cloud fog offloading," in 2019 15th IEEE International Workshop on Factory Communication Systems (WFCS), 2019: IEEE, pp. 1-4.

[16] B. Pourghebleh and N. J. Navimipour, "Data aggregation mechanisms in the Internet of things: A systematic review of the literature and recommendations for future research," Journal of Network and Computer Applications, vol. 97, pp. 23-34, 2017.

[17] A. Kumar et al., "Optimal cluster head selection for energy efficient wireless sensor network using hybrid competitive swarm optimization and harmony search algorithm," Sustainable Energy Technologies and Assessments, vol. 52, p. 102243, 2022.

[18] M. Mohseni, F. Amirghafouri, and B. Pourghebleh, "CEDAR: A cluster-based energy-aware data aggregation routing protocol in the internet of things using capuchin search algorithm and fuzzy logic," Peer-to-Peer Networking and Applications, pp. 1-21, 2022.

[19] B. Pourghebleh, K. Wakil, and N. J. Navimipour, "A comprehensive study on the trust management techniques in the Internet of Things," IEEE Internet of Things Journal, vol. 6, no. 6, pp. 9326-9337, 2019.

[20] B. Pourghebleh and V. Hayyolalam, "A comprehensive and systematic review of the load balancing mechanisms in the Internet of Things," Cluster Computing, pp. 1-21, 2019.

[21] A. Mehbodniya, J. L. Webber, R. Neware, F. Arslan, R. V. Pamba, and M. Shabaz, "Modified Lamport Merkle Digital Signature blockchain framework for authentication of internet of things healthcare data," Expert Systems, vol. 39, no. 10, p. e12978, 2022.

[22] A. Kumar, S. A. Alghamdi, A. Mehbodniya, J. L. Webber, and S. N. Shavkatovich, "Smart power consumption management and alert system using IoT on big data," Sustainable Energy Technologies and Assessments, p. 102555, 2022.

[23] H. Hindy et al., "A taxonomy of network threats and the effect of current datasets on intrusion detection systems," IEEE Access, vol. 8, pp. 104650-104675, 2020.

[24] M. Ge, N. F. Syed, X. Fu, Z. Baig, and A. Robles-Kelly, "Towards a deep learning-driven intrusion detection approach for Internet of Things," Computer Networks, vol. 186, p. 107784, 2021.

[25] V. Hayyolalam, B. Pourghebleh, and A. A. Pourhaji Kazem, "Trust management of services (TMoS): Investigating the current mechanisms," Transactions on Emerging Telecommunications Technologies, vol. 31, no. 10, p. e4063, 2020.

[26] A. Mehbodniya, J. Webber, K. Yano, T. Kumagai, and M. F. Flanagan, "Gibbs Sampling Aided Throughput Improvement for Next-Generation Wi-Fi," in 2018 IEEE Globecom Workshops (GC Wkshps), 2018: IEEE, pp. 1-6.

[27] P. A. A. Resende and A. C. Drummond, "Adaptive anomaly-based intrusion detection system using genetic algorithm and profiling," Security and Privacy, vol. 1, no. 4, p. e36, 2018.

[28] T. A. Alamiedy, M. Anbar, Z. N. Alqattan, and Q. M. Alzubi, "Anomaly-based intrusion detection system using multi-objective grey wolf optimisation algorithm," Journal of Ambient Intelligence and Humanized Computing, vol. 11, no. 9, pp. 3735-3756, 2020.

[29] Y. Zhang, P. Li, and X. Wang, "Intrusion detection for IoT based on improved genetic algorithm and deep belief network," IEEE Access, vol. 7, pp. 31711-31722, 2019.

[30] A.-u.-H. Qureshi, H. Larijani, J. Ahmad, and N. Mtetwa, "A heuristic intrusion detection system for Internet-of-Things (IoT)," in Intelligent computing-proceedings of the computing conference, 2019: Springer, pp. 86-98.

[31] S. Dwivedi, M. Vardhan, S. Tripathi, and A. K. Shukla, "Implementation of adaptive scheme in evolutionary technique for anomaly-based intrusion detection," Evolutionary Intelligence, vol. 13, no. 1, pp. 103-117, 2020.

[32] R. Gassais, N. Ezzati-Jivan, J. M. Fernandez, D. Aloise, and M. R. Dagenais, "Multi-level host-based intrusion detection system for Internet of things," Journal of Cloud Computing, vol. 9, no. 1, pp. 1-16, 2020.

[33] S. Aldhaheri, D. Alghazzawi, L. Cheng, B. Alzahrani, and A. Al-Barakati, "Deepdca: novel network-based detection of iot attacks using artificial immune system," Applied Sciences, vol. 10, no. 6, p. 1909, 2020.

[34] J. Brown and M. Anwar, "Blacksite: human-in-the-loop artificial immune system for intrusion detection in internet of things," Human-Intelligent Systems Integration, vol. 3, no. 1, pp. 55-67, 2021.

[35] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," Cybersecurity, vol. 2, no. 1, pp. 1-22, 2019.

[36] O. Nguyen, "HSSCIoT: An Optimal Framework based on Internet of Things-Cloud Computing for Healthcare Services Selection in Smart Hospitals," Advances in Engineering and Intelligence Systems, vol. 1, no. 02, 2022.

[37] F. Kuang, W. Xu, and S. Zhang, "A novel hybrid KPCA and SVM with GA model for intrusion detection," Applied Soft Computing, vol. 18, pp. 178-184, 2014.

[38] W. Wang, X. Zhang, and S. Gombault, "Constructing attribute weights from computer audit data for effective intrusion detection," Journal of Systems and Software, vol. 82, no. 12, pp. 1974-1981, 2009.

[39] A. Al-Moalmi, J. Luo, A. Salah, K. Li, and L. Yin, "A whale optimization system for energy-efficient container placement in data centers," Expert Systems with Applications, vol. 164, p. 113719, 2021.