An Intelligent Approach for Intrusion Detection using Snake Optimizer and Random Forest

Salahaldeen Duraibi
Department of computer and network engineeringy, Jazan University
Jazan, Saudi Arabia
sduraibi@jazanu.edu.sa

***Abstract*—In recent years, many researchers have used various Machine Learning (ML) models that have demonstrated the power of such methods on Intrusion Detection (ID) and thus helped classifying network packets as normal system behavior or an attack. This paper presents a novel SO-RF model that combines Snake Optimizer (SO) and Random Forest (RF) for ID. The SO Meta-Heuristics (MH) algorithm is employed to select Optimal Feature Subset (OFS) from large datasets and the resulted OFS is used by the RF model to improve learning process and classification accuracy. The SO-RF is validated on two datasets for ID: KDD CUP99 and NSL-KDD. Results show that the introduced SO-RF achieves better performance outcomes compared to the RF, SVM, and several other reported models in the literature for ID.**

Keywords—intrusion detection system, machine learning, metaheuristic algorithms, feature selection

I. Introduction

The widespread computer networks increased internet usage rate caused, and the security of such networks is one of the most critical research areas as threats and attacks on these networks become more and more aggressive than before [1]. Several security technologies, such as firewalls, aauthentication, and encryption, are employed to deal with and prevent many attacks. Despite the powerful capabilities of these technologies, several attacks derived from these slow-developing technologies remain undetected and are successful in penetrating these networks. Intrusion Detection System (IDS) and Intrusion Prevention System (IPS) are developed to perform deeper data analysis and overcome the shortcomings of earlier security systems.

Due to the adaptive forms of attacks in terms of viruses, bots, threats, and malware, cybersecurity companies focus on producing more sensitive systems in addition to the traditional security methods [2–4]. On the other side, proactive cyber-security systems such as, network behavior analysis and threat analysis are developed. IDS is one of the frequently used technology that have become more vital for cyber-security. It is is a software package that is responsible for detecting threats across the network or system

Researchers explored Machine learning (ML) approaches for achieving networks' optimal security requirements to develop an Intrusion Detection (ID) model that can detect such attacks with high accuracy [5,6]. ML approaches for ID gained special attention because of its ability to use hundreds of features for classifying normal system behavior and attack attempts [7, 8]. The primary purpose of Feature Selection (FS) as a technique is to select the Optimum Feature Subset (OFS) in a given dataset for optimizing the learning process.

Several earlier studies have used Meta-Heuristic (MH) methods in ID [9, 10, 11, 12]. In [13], the authors presented an intelligent approach for ID called DRCNN-IDS using Convolutional Neural Network (CNN) with dimensionality reduction on the KDD-Cup99 dataset. The DRCNN-IDS is 96% accurate and performed better than the other ML methods used in their work. In another work [14], six ML models comprising J48, Random Forest (RF), Random Tree (RT), Multi-Layered Perceptron (MLP), Naïve Bayes (NB), and Bayes Network (BN) are used for ID. The RF reported the best performance with an accuracy of 93.77% for the KDD-Cup99 dataset.

In another work [15], multiple FS methods are employed to choose OFS as input to the Support Vector Machine (SVM) for ID. They evaluated their model using KDD CUP99 and NSL-KDD benchmark datasets. The experimental results reported accuracy of 98.95% using the KDD-CUP99 dataset and 98.12% using the NSL-KDD dataset. In another work [16], the Oppositional Crow Search Algorithm (OCSA), an integration of the Crow Search Algorithm (CSA) and Opposition Based Learning (OBL) method, is used to select OFS input to Recurrent Neural Network for ID. The results indicate superior performance than earlier IDS with an accuracy of 94.12% using KDD-CUP99 dataset.

In [17], Deep Belief Network (DBN) and SVM advantages are combined for ID. The DBN is employed to select the most informative features and SVM to classify intrusion into normal or attack attempts. The results show an overall accuracy of 92.84% using the NSL-KDD dataset. In [18], several ML models are analyzed using Particle Swarm Optimization (PSO) based FS for ID. The results show that PSO with Neural Network (NN) achieved the best accuracies of 99.20% and 99.65% using KDD-CUP99 and using the NSL-KDD datasets, respectively.

The paper introduces a novel IDS by combining Snake Optimizer (SO) and Random Forest (RF), named SO-RF. The proposed SO-RF uses SO for searching OFS and RF for building a classification model. The contributions of the work are as follows:

- A novel SO-RF model with the SO-based FS method to reduce feature dimensionality and RF-based classification model to increase performance for ID is developed.
- The effectiveness of the SO-RF approach is investigated using several quantitative evaluation measures on KDD-CUP99 and NSL-KDD datasets.
- The quantitative evaluations show better performance of the SO-RF model than RF, SVM, and other earlier reported models in the literature for ID.

The rest of this paper is structured as follows: In section 2, a briefly overview for the SO, RF and SO-RF for ID is described. In section, 3, experimental results, datasets description, , evaluation metrics and discussion are presented.

979-8-3503-2028-2/22/$31.00 ©2022 IEEE
DOI 10.1109/CSCI58124.2022.00022

Section 4, the concludes the work with few future research directions.

II. Proposed Method

A. Snake Optimizer (SO)

SO is a new Meta-Heuristic (MH) optimization reported in the literature that mimics the snakes' mating in presence of food and low temperature is low [19]. Similar to general MH algorithms, SO generates populations to begin the randomization process as:

$$x_i = \ x_{min} \ + r\ (x_{max} - x_{min}) \tag{1}$$

where, x_i is the position of ith individual, r is a random number in the range of [0,1], and x_{max} and x_{min} represent the upper and lower boundaries.

The entire snake population is equally split into male and female subgroups using the following:

$$N_m \approx N/2 \ \text{ and } \ N_f = N - \ N_m \tag{2}$$

where, N is the number of snakes, N_m refers to the male individual numbers and N_f refers to the individual female snakes.

Find the best individual in each group and get the best male,$f_{best,m}$ best female $f_{best,f}$ and food position f_{food}. The Temperature (T) at the current iteration (g) and available Food Quantity (FQ) are calculated as:

$$T = exp\left(\frac{-g}{T}\right) \text{ and } FQ = c_1 exp\left(\frac{g-T}{T}\right) \tag{3}$$

where, T is the total number of iterations and c_1 equals to 0.5.

When FQ is less than threshold of 0.5, snakes explore by updating their positions to continue food search. The analytical model for this behavior is as follows:

- Male snakes:

$$x_{i,m}(g+1) = x_{rand,m}(g) \pm c_2 \times A_m\big((x_{max} - x_{min}) \times rand + x_{min}\big) \tag{4}$$

where $A_m = exp\left(\frac{-f_{rand,m}}{f_{i,m}}\right)$

where, $x_{i,m}$ is ith male position, $x_{rand,m}$ is a random male snake position, $rand$ is a random number between 0 and 1, $A_m$is male snake's food finding ability, $f_{rand,m}$ is the fitness of $x_{rand,m}$ and $f_{i,m}$ is the fitness of ith individual in the male group. A random flag direction operator $\pm$ facilitates search space scanning in all the possible directions.

- Female snakes:

$$x_{i,f} = x_{rand,f}(g+1) \pm c_2 \times A_f\big((x_{max} - x_{min}) \times rand + x_{min}\big) \tag{5}$$

where $A_f = exp\left(\frac{-f_{rand,f}}{f_{i,f}}\right)$

where, $x_{i,f}$ is ith female position, $x_{rand,f}$ is the position of random female, A_f refers to her ability to find the food, $f_{rand,f}$ is the fitness of $x_{rand,f}$ and $f_{i,f}$ is ith female's fitness.

Two conditions are used in SO exploitation phase to find the best solutions are:

1. If FQ < Threshold (T > 0.6), then the snakes move to the find only:

$$x_{i,j}(g+1) = x_{food} \pm c_3 \times T \ \times rand \ \times \big(x_{food} - x_{i,j}(g)\big) \tag{6}$$

where, $x_{i,j}$ is male or female position, x_{food} is the best snake's position and c_3 is a constant equal to 2.

2. If FQ < Threshold (Threshold < 0.6), then the snakes will be either fighting or matting, as follows:

- Fighting mode

The male F_{male} and female F_{female} agents fighting mode can be written as:

$$x_{i,m}(g+1) = x_{i,m}(g) \pm c_3 \times F_{male} \ \times rand \ \times \big(x_{best,f} - x_{i,m}(g)\big), \tag{7}$$

where, $F_{male} = exp\left(\frac{-f_{best,f}}{f_i}\right)$

where. $x_{i,m}$, refers to ith male position, $x_{best,f}$refers to the best female's position and male's fighting ability F_{male}.

$$x_{i,f}(g+1) = x_{i,f}(g) \pm c_3 \times F_{female} \ \times rand \ \times \big(x_{best,m} - x_{i,f}(g+1)\big) \tag{8}$$

where $F_{female} = exp\left(\frac{-f_{best,m}}{f_i}\right)$

where. $x_{i,f}$, refers to ith female position, $x_{best,m}$refers to the best male's position and F_{female} is the female's fighting ability.

- Mating Mode:

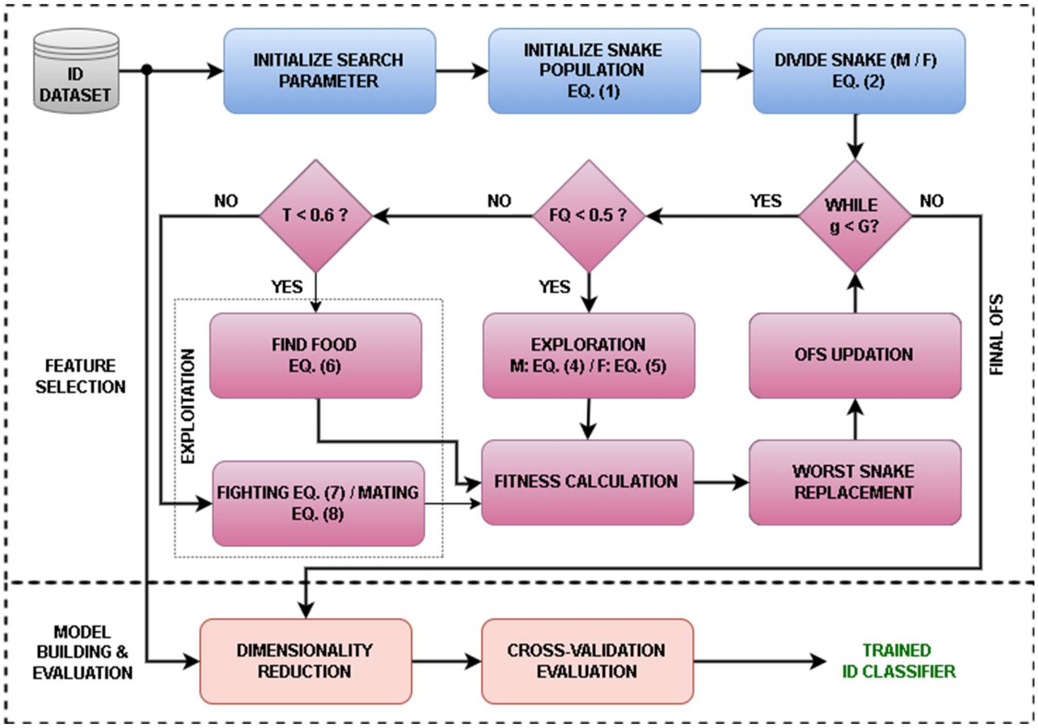

ID DATASET
INITIALIZE SEARCH PARAMETER
INITIALIZE SNAKE POPULATION EQ. (1)
DIVIDE SNAKE (M / F) EQ. (2)
NO
T < 0.6 ?
NO
FQ < 0.5 ?
YES
WHILE g < G?
NO
YES
YES
FEATURE SELECTION
EXPLOITATION
FIND FOOD EQ. (6)
EXPLORATION M: EQ. (4) / F: EQ. (5)
OFS UPDATION
FINAL OFS
FIGHTING EQ. (7) / MATING EQ. (8)
FITNESS CALCULATION
WORST SNAKE REPLACEMENT
MODEL BUILDING & EVALUATION
DIMENSIONALITY REDUCTION
CROSS-VALIDATION EVALUATION
TRAINED ID CLASSIFIER

Fig. 1. Flow diagram of the SO-RF model

Analytically, mating mode of the male and female snakes is as follows:

$$x_{i,m}(g+1) = x_{i,m}(g) \pm c_3 \times M_{male} \times rand \times \left(Q \times x_{i,f} - x_{i,m}(g)\right) \qquad (8)$$

where $M_{male} = exp\left(\frac{-f_{i,f}}{f_{i,m}}\right)$

$$x_{i,f}(g+1) = x_{i,f}(g) \pm c_3 \times M_{female} \times rand \times \left(Q \times x_{i,m} - x_{i,f}(g+1)\right),$$

where $M_{female} = exp\left(\frac{-f_{i,m}}{f_{i,f}}\right)$

where, $x_{i,m}$ and $x_{i,f}$is the position of ith agent of male and female group, M_{male} and M_{female} refer to the mating ability of male and female

Select worst male and female and then replace them using the following:

$$x_{worst,m} = x_{min} + rand + (x_{max} - x_{min})$$

$$x_{worst,f} = x_{min} + rand + (x_{max} - x_{min}) \qquad (10)$$

where, $x_{worst,m}$ and $x_{worst,f}$ represent the worst individual in male and female respectively.

B. Random Forest (RF)

RF was introduced in [20] for solving both regression and classification problems. It is an ensemble classifier with small training speed, suitability in scientific and engineering applications, and complex datasets capability [21]. RF uses majority voting based aggregation of results from many non-pruned Decision Trees (DT). The diversity of treess is increased by generating each DT from bootstrap data drawn from the training data. The samples not involved in the generating DTs are known as 'Out-Of-Bag' (OOB) data. During training phase, RF uses OOB data internally for validation. The RF prediction model can be presented as [20]:

$$f_{rf}^{N}(x) = \frac{1}{N}\sum_{n=1}^{N} T_{ree}(x), \qquad x = x_1, x_2, \cdots x_p \quad (11)$$

where N is the average number of regression trees in RF, x is a p-dimensional input vector and T_{ree} denotes DT.

C. Proposed SO-RF

A combination of SO algorithm for OFS selection and RF for classification is presented in this section. The randomness in SO decreases the possibility being restrained to local optimum solution by reducing redundant feature dimension. The reduced dimensionality aids RF in arranging features in OFS to maximize the detection performance without much confusion.

The work flow of the proposed SO-RF model is displayed in Figure 1. It can be understood in two phases: (i) feature selection and (ii) model building and evaluation. In the first step, complete feature set in passed to SO to generate OFS. The process starts with initialization of search parameters of SO that comprises algorithm constants like $c_1, c_2,$ and c_3, number of expected solutions or snakes in SO (N), and maximum number of iteration G. Also, feature dimensionality constant M and extremeties of each dimension x_{min} (lower

TABLE I. PARAMETERS OF THE PROPOSED SO-RF MODEL.

Parameter name	Value
SO constants (c_1, c_2, c_3)	(0.5, 0.05, 2)
Lower x_{min} and upper x_{max} bounds	as per dataset
Maximum number of iterations (N)	100
Feature dimensionality (M)	as per dataset
Maximum number of DTs	500
Minimum number of features at each node	8
Minimum population of lean node	5

bound) and x_{max} (upper bound) are calculated from the dataset. Initial positions of N snakes in M-dimension space is randomized uniformly in the range [-1, 1], as described earlier in Eq. (1). This population is randomly divided in male and female snakes, as described earlier in Eq. (2).

At each iteration, positions of each snake $\{x_{i,j}(g), 1 \leq i \leq N \,\&\, 1 \leq j \leq M\}$ are updated by following exploration and exploitation steps, as described earlier in Eq. (4)–(8). At the end of each iteration, the weak snakes identified based on minimum Fitness Value (FV) are replaced by best snakes of the same gender. Finally, OFS is updated based on positions of the snake with least FV. A feature is added in OFS is its current position is greater than 0.5 else it will be rejected. It can well be noted that threshold of 0.5 is indicative value and over sufficiently large number of runs does not affect the optimization.

The updated snake positions are given as input to the next iteration and the process repeats itself until maximum number of iterations. The final OFS at the end of feature selection is shared to model building and evaluation phase. In this phase, firstly the dimensionality of the input dataset is reduced by selecting only the features present in the OFS. The selected features are given as input to the classifier. The classifier is trained using cross-validation technique to tune its hyper-parameters and maximize the detection performance in general sense. The cross-validation also alleviates the chances of overfitting which can easily occur in IDS due to overwhelming size of the dataset. At the end of this phase a trained classifier is output which can be used to classify normal activities and attack attempts. The hyper-parameters used for SO-RF training are denoted in Table 1.

III. EXPERIMENTAL RESULTS

A. Experimental setup

The SO-RF is evaluated by performing experiments on two standardized and popular datasets in the area of ID: KDD-CUP99 and NSL-KDD datasets. The KDD-CUP99 dataset includes DoS, Remote to Local (R2L), User to Root (U2R), and probing attack properties. It consists of seven weeks of network traffic, it has about 5 million lines, and is one of the most widely used datasets for assessing ID models. The NSL-KDD is an upgraded version of KDD- CUP99, that includes 42-dimensional feature in each record. It does not contain un-necessary and repetitive records according to the original KDD- CUP99 dataset and uses the same properties as the KDD-CUP99 [22].

To avoid possible bias in selecting the training and testing datasets, the 10-fold Cross-validation (CV) method is employed. All the experiments are implemented in Python and executed on a 3.13 GHz PC with 16 GB RAM and Windows 10 operating system. The description of those datasets is presented in Table 2.

TABLE III. KDD- CUP99 AND NSL-KDD DATASETS

Dataset	Year	No. of features	No. of samples
KDD-CUP99	1998	43	494020
NSL-KDD	2009	43	125973

B. Evaluation measures

Various measures can be used to evaluate proposed SO-RF model efficiency. The accuracy (AC), precision (P) , Recall (R), and F1-measure are used, and they are calculated as follows:

$$AC = \frac{TP+TN}{TP+TN+FN+F} \tag{12}$$

$$\text{Precision } (P) = \frac{TP}{TP+FP} \tag{13}$$

$$\text{Recall } (R) = \frac{TP}{TP+FN} \tag{14}$$

$$\text{F1–score } (F) = \frac{2\,P\,R}{P+R} \tag{15}$$

where, True Positive and (TP) and True Negative (TN) denote the samples of customers correctly detected as churner or not, while False Negative (FN) and False Positive (FP) represents the number of misclassified positive and negative cases, respectively.

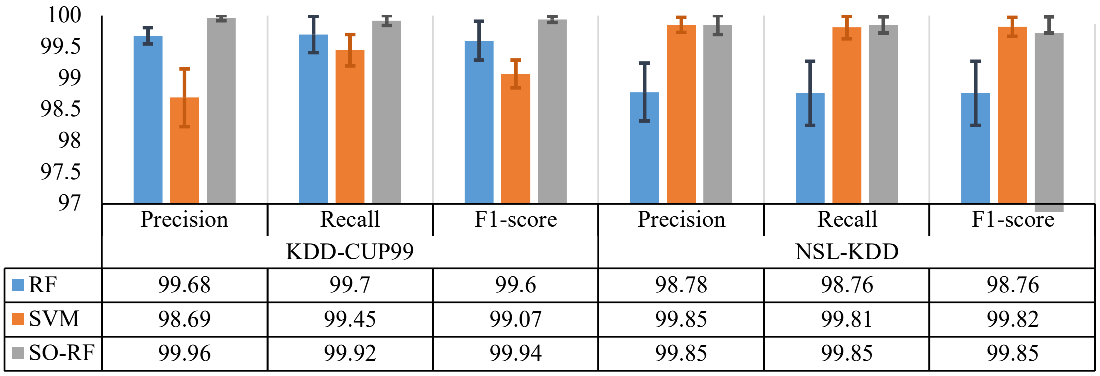
C. Experimental results and discussion

In order to examine the effectiveness of the SO-RF model, the real-world datasets provided in Table 1 are used. Table 3 gives the mean and Standard Deviation (SD) of accuracies for RF, SVM, and proposed SO-RF models using KDD-CUP99 and NSL-KDD datasets. The higher mean accuracy of the proposed SO-RF model than RF indicates that FS using SO has boosted the overall accuracy of the RF model by reducing the redundant features during classification decisions. The proposed SO-RF has also achieved higher accuracy than SVM, which shows that the FS by SO has not affected the features integral for ID. The SD of the proposed SO-RF model is the smallest among the three, demonstrating more stability than the remaining two models.

Figure 2 shows the comparative analysis of proposed SO-RF, RF, and SVM based IDS using remaining three quantitative evaluation measures for both datasets. The type of evaluation measure is plotted on horizontal axis with values on vertical axis and different IDSs are marked by different colours. Figure 2 shows that the proposed system has highest precision using both datasets while recall is highest for NSL-

TABLE II. COMPARATIVE ANALYSIS IN TERM OF ACCURACY FOR THE SO-RF, RF, AND SVM MODELS USING KDD-CUP99 AND NSL-KDD DATASETS FOR ID

Model		KDD-CUP99	NSL-KDD
RF	Mean	99.70%	98.76%
	SD	0.4506	0.6643
SVM	Mean	98.89%	99.85%
	SD	0.4690	0.4642
Proposed SO-RF	Mean	99.97%	99.83%
	SD	0.4376	0.3513



	Precision	Recall	F1-score	Precision	Recall	F1-score
	KDD-CUP99			NSL-KDD		
🟦 RF	99.68	99.7	99.6	98.78	98.76	98.76
🟧 SVM	98.69	99.45	99.07	99.85	99.81	99.82
⬜ SO-RF	99.96	99.92	99.94	99.85	99.85	99.85

Fig. 2. Quantitative comparison of the proposed SO-RF, RF, and SVM models using KDD-CUP99 and NSL-KDD datasets for ID

TABLE IV. ACCURACY COMPARISON BETWEEN THE EARLIER REPORTED MODELS AND THE PROPOSED SO-RF.FOR ID

Model	KDD CUP99	NSL-KDD
DRCNN-IDS, [13]	96%	-
RF [14]	93.77%	-
SVM [15]	98.95%	98.12%
OCSA-RNN [16]	94.12%	-
DBN-SVM [17]	-	92.84%
PSO-NN [18]	99.20%	99.65%
Proposed SO-RF	99.97%	99.83%

KDD. Although, the precision of the proposed SO-RF is equal to SVM for KDD-CUP99, the F1-score is of the proposed SO-RF is slightly higher than remaining two.

D. *Comparison with existing models*

Recently, several works have been proposed for ID. The studies in Table 4 used KDD CUP99 and NSL-KDD datasets to validate their proposed model's efficiency. As per the results in Table 4, the proposed SO-RF model provides higher accuracy for both KDD CUP99 and NSL-KDD datasets than the existing models.

IV. CONCLUSION AND FUTURE WORKS

In this paper, SO-RF model is developed and presented for ID, the SO-RF uses SO for feature selection and RF as a learning model. Two datasets are used to test the efficacy of the SO-RF model and they include KDD CUP99 and NSL-KDD. Results show that the introduced SO-RF achieves better results compared to RF and SVM models in terms of several evaluation measures. Moreover, the SO-RF is superior to other recent reported approaches for ID in the literature. In future, will plan to use proposed SO-RF model in different applications such as renewal energy, signal processing and big data. Another possible avenue is to work on MH methods or nature- inspired algorithm for FS in the application of ID, because these optimization algorithms have also shown excellent results in other domains

REFERENCES

[1] I. H. Sarker, A. S. M. Kayes, S. Badsha, H. Alqahtani, P. Watters, & A. Ng, "Cybersecurity data science: an overview from machine learning perspective," J. Big data, vol. 7, no. 1, pp. 1–29, 2020.

[2] E. Jaw & X. Wang,. "Feature selection and ensemble-based intrusion detection system: an efficient and comprehensive approach," Symmetry, vol. 13, no. 10, 1764, 2021.

[3] H. Kure & S. Islam, "Cyber threat intelligence for improving cybersecurity and risk management in critical infrastructure," J. Universal Comput. Sci., vol. 25, no. 11, pp. 1478–1502, 2019.

[4] I. Lee, "Internet of Things (IoT) cybersecurity: Literature review and IoT cyber risk management," Future Internet, vol. 12, np. 9, 157, 2020.

[5] V. Ford & A. Siraj, "Applications of machine learning in cyber security," in Proc. 27th Int. Conf. Comput. Appl. Industry and Engg., Kota Kinabalu, Malaysia, vol. 118, 2014,

[6] A. Gupta, R. Gupta, & G. Kukreja, "Cyber security using machine learning: techniques and business applications," in Appl. Artificial Intel. Bus., Edu. Healthcare, pp. 385–406, 2021.

[7] Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah, & F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," Trans. Emerging Telecommun. Technolo., vol. 32, no. 1, e4150, 2021.

[8] H. Liu & B. Lang, "Machine learning and deep learning methods for intrusion detection systems: A survey," Applied Sci., vol. 9, no. 20, 4396, 2019.

[9] A. Shenfield, D. Day, & A. Ayesh, "Intelligent intrusion detection systems using artificial neural networks," ICT Express, vol. 4, no. 2, pp. 95–99, 2018.

[10] N. Bashah, I. B. Shanmugam, & A. M. Ahmed, "Hybrid intelligent intrusion detection system" World Academy of Sci. Engg. Technolo., vol. 11, pp. 23–26, 2005.

[11] M. Almi'ani, A. A. Ghazleh, A. Al-Rahayfeh, & A. Razaque, "Intelligent intrusion detection system using clustered self organized map," in Proc. Fifth Int. Conf. Softw. Defined Syst. (SDS), pp. 138–144, 2018.

[12] R. V. Mendonça, J. C. Silva, R. L. Rosa, M. Saadi, D. Rodriguez, D. Z., & A. Farouk, "A lightweight intelligent intrusion detection system for industrial internet of things using deep learning algorithms," Expert Syst., vol. 39, no. 5, e12917, 2022.

[13] V. Manikandan, K. Gowsic, T. Prince, R. Umamaheswari, B. F. Ibrahim, & A. Sampathkumar, "DRCNN-IDS approach for intelligent intrusion detection system," in Int. Conf. Comput. Info. Technolo., pp. 1–4, 2020.

[14] I. Obeidat, N. Hamadneh, M. Alkasassbeh, M. Almseidin, & M. AlZubi, "Intensive pre-processing of kdd CUP99 for network intrusion classification using machine learning techniques", pp. 70-84, 2019.

[15] A. K. Shukla & P. Singh "Building an effective approach toward intrusion detection using ensemble feature selection," Int J Info Sec Privacy, vol. 13, no. 3, pp. 31–47, 2019.

[16] R. SaiSindhuTheja & G. K. Shyam, "An efficient metaheuristic algorithm based feature selection and recurrent neural network for DoS attack detection in cloud computing environment," Applied Soft Comput, vol. 100, 106997, 2021.

[17] M. A. Salama, H. F. Eid, R. A. Ramadan, A. Darwish, & A. E. Hassanien, "Hybrid intelligent intrusion detection scheme," Soft Comput Industrial Appl., pp. 293–303, 2011.

[18] S. Khan, K. Kifayat, A. Kashif Bashir, A. Gurtov, & M. Hassan, "Intelligent intrusion detection system in smart grid using

computational intelligence and machine learning," Trans Emerging Telecommun Technolo, vol.32, no. 6, e4062, 2021.

[19] F. A. Hashim & A. G. Hussien, "Snake Optimizer: A novel meta-heuristic optimization algorithm," Knowledge-Based Syst., vol. 242, 108320, 2022.

[20] R. Jiang, W. Tang, X. Wu, & W. Fu, "A random forest approach to the detection of epistatic interactions in case-control studies," BMC bioinformatics, vol. 10, no. 1, pp. 1–12, 2019.

[21] R. Genuer, J. M. Poggi, & C. Tuleau-Malot, "Variable selection using random forests," Pattern Recognition Lett., vol. 31, no. 14, pp. 2225–2236, 2010.

[22] S. Sapre, P. Ahmadi, & K. Islam, "A robust comparison of the KDDCup99 and NSL-KDD IoT network intrusion detection datasets through various machine learning algorithms," arXiv preprint arXiv:1912.13204, 2019.