Received: September 3, 2024. Revised: October 1, 2024.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

An Accurate Approach for Intrusion Detection System Using Chaotic Maps, NPO, and SVM

Zinah Sattar Jabbar Aboud[1,2]* Rami Tawil[1] Mustafa Salam Kadhm[3]

[1]*The Lebanese University, Lebanon*
[2]*Department of Computer Technology Engineering, College of Information Technology, Imam Ja'afar Al-Sadiq University, Baghdad, Iraq*
[3]*Computer Department, College of Basic Education, Mustansiriyah University, Iraq*
* Corresponding author's Email: sattarzeina@gmail.com

Abstract: The internet and technological advancements have facilitated faster communication and information sharing. However, cybercrime, including malware, phishing, and ransomware, remains a severe problem despite technical progress. Detecting the intrusion via Intrusion Detection System IDS in network communication and wireless networks WSN is a big challenge that grown with the rapid development of the technologies. The detection accuracy of the IDS mainly depends on the relevant features of the incoming data from the internet. Selecting the most relevant features within the optimal attributes is one of the primary stage of the machine learning and pattern recognition modules. Finding the feature subset from the present or existing features that will improve the algorithms' learning performance in terms of accuracy and learning time is the main goal of feature selection. Therefore, this paper proposes an accurate approach for intrusion detection in the network and WSN using machine learning methods include Chaotic Maps, Nomadic People Optimizer (NPO), and SVM. The proposed approach has five main stages which are: data collection, pre-processing, feature selection, classification, and evaluation. An improved version of NPO based on chaotic map called CNPO is proposed. The proposed CNPO uses chaotic maps to initialize the population and solution distribution. Besides, a proposed fitness function for CNPO based on SVM is proposed. The CNPO is employed for feature selection task by selecting only the most relevant features from the input dataset. The proposed approach evaluated using two datasets and achieve accuracy 99.96% and 99.98 for NSL-KDD, and WSN-DS respectively.

Keywords: Intrusion detection, NPO, CNPO, Chaotic, SVM.

1. Introduction

The rapid growth of network and communication fields, as well as the increase in the amount of data carried via these networks, has exacerbated the security problem for these data. The internet poses a serious thread on the user privacy and the data security [1]. Users on internet suffer from several types of cyber-attacks that lead to losing their personal data expose them to various types of blackmail and cybercrimes [2]. In order to prevent all possible attacks in the internet a capable system that detect anomalies and protecting the network called Intrusion Detection Systems (IDS) is presented [3].

IDS is a critical part of the cybersecurity that designed to detect the unauthorized access in a computer network or a computer system. IDS monitor the network traffic, and system activities to identify any possible threats such as: suspicious behaviour, insider threats, and malware. When unauthorized behaviour is detected by IDS, the system will immediately create an alert so that any further attacks can be mitigated and prevented [4-6].

IDS has two main detection methods which are: signature based detection and anomaly based detection [7]. Signature based detection compares the incoming data from the internet with a database of known attack signatures to identify the threats. While, the anomaly based detection identifies the threats via establishing a baseline of the normal behaviour, when

there is anomalous behaviour throughout the bassline, the system will detect it as a potential threat [8, 9].

Machine learning methods are utilized to improve the IDS's performance [10]. The machine learning enhancing the capabilities of IDS by addressing the limitation of the traditional methods, via enhancing the IDS accuracy, efficiency, and adaptability. Several researches conducted for IDS focused on improving detection results using classification methods and feature selection methods in machine learning [11].

Feature selection is used in IDS by identifying the most relevant features from the used dataset that contribute to the accurate detection of the intrusion. Feature selection improves the IDS performance, efficiency, and interpretability by selecting only the most informative features. In general, there are four types of feature selection techniques for IDS include: filters, wrappers, embedded selectors, and dimensionality reduction [12]. Number of researches conducted for improving the IDS performance using various feature selection methods [13-17].

Researchers in [18, 19] used cuckoo algorithm and improved cuckoo algorithm for feature selection process. Also in [20] Modified Metaheuristics with Weighted Majority Voting Ensemble Deep Learning (MM-WMVEDL) model is used for IDS. While in [21] Explored Particle Swarm Optimization (PSO) cantered Sea Turtle Foraging Algorithm (EXPSO-STFA) are employed for feature selection. Other optimization algorithms in [22-25] were effectively applied to IDS as a feature selector resulting in satisfactory detection results across a variety of IDS datasets.

NPO is a metaheuristic algorithm that inspired by the life style of the nomadic communities in the desert searching for the useful life sources to move toward it [26]. NPO designed based on the multi-swarm approach with several clans for finding the best solution via following the leader position. NPO is employed in various researches and obtained the best optimization results [27-28].

The paper proposes an accurate intrusion detection system using an improved version of NPO algorithm called CNPO. The NPO initially enhanced using logistic and circle chaotic maps for improving the search space of the solution and create an optimal diversity. Furthermore, a proposed fitness function based on SVM is applied in NPO. SVM is additionally utilized for attacks detection and classification. The CNPO is evaluated using two standard intrusion detection datasets which are NSL-KDD, and WSN-DS. The main contributions of the proposed system are listed below:

- The NPO is improved via the position update equation.
- Logistic map is used for improve NPO via initialize the population.
- Circle map is used for improve NPO via distribution process.
- A proposed fitness function based on SVM is included in CNPO.
- The update equation of NPO is improved.
- SVM with RBF kernel is used for classification
- The proposed system evaluated using two datasets NSL-KDD, and WSN-DS.
- The obtained results of the proposed system are compared with the most recent works in intrusion detection

The rest of the paper structured as following: Section 2 describe the literature review and the proposed system include NPO, Chaotic maps and SVM is described in Section 3. The obtained results in Section 4. Last, in Section 5 the conclusion is stated.

2. Literature review

Several works have been done for intrusion detection for both network and WSN. The recent works focused on detecting attacks using various machine learning methods. Some works used different classifiers and deep learning for better detection results. However, other works employed optimization algorithms for feature selection process, thereby improving the detection performance. The following is a brief summary of recent intrusion detection research.

Voya et al. (2024) presented a comprehensive analysing of the intrusion detection system using several machine learning algorithms which are Support Vector Machine (SVM), Decision Tree (DT), and Random Forest (RF). The work aims to evaluate the performances of the algorithms using NSL-WSN dataset in various scenarios. Supervised, unsupervised, and semi-supervised algorithms are used in the analysis. SVM demonstrates superior performance in intrusion detection results comparing to DT and RF. The highest accuracy was achieved by SVM 95.2%, with precision 92.6%, recall 94.3%, and F1-score 93.4%. However, DT and RF achieved an accuracy 92.7% and 94.5% respectively. The work still suffers from dataset imbalance, interpretability, adversarial robustness, and low detection accuracy [29].

N. Girubagari et al. (2023) proposed an algorithm for real-time intrusion detection system called PACENIDS using ensemble of Altered Bi-directional Long Short-Term Memory (ABILSTM) and

Customized Bi-directional Gated Recurrent Unit (CBIGRU). The proposed algorithm is employed to detect the attacks in the smart cities networks. In order to improve the performance of the proposed algorithm, authors used fuzzy feature selection algorithm. PACENIDS achieved a high classification accuracy 96.59%, 94.47% without feature selection algorithm, and 97.67% classification accuracy with feature selection algorithm using NSL-KDD dataset. However, the applied convolutional architecture takes more time to train the system, which the main limitation of the work [30].

Samer et al. (2023) proposed a hybrid filter-wrapper feature selection method for intrusion detection system called GBA. The proposed method selects a feature subset from the original features to improve the performance of the system. The filter feature selection is based on Information Gain (IG) algorithm, and the wrapper feature selection is based on the Black Hole (BH) algorithm. The aim of GBA to improve the accuracy of the IDs by initialize the features for classification using IG by ignore the zero weighted features. GBA achieved a classification accuracy 96.96% using NSL-WS dataset. However, the work used only one dataset and the obtained accuracy can be improved further [31].

Akindele S. et al (2024) presented combination approach between optimization and machine learning algorithms for network intrusion detection called KOMIC IDS. knapsack optimization algorithm (KO) used first to select the relevant features from the IDs dataset with mutual information gain filter (MIC). After that, a new set of features combine with the selected features. MIC is applied again on the combined features to remove the duplicated features and keep only the highest information gain features. In another hand, several machine learning classifiers are used to evaluate the performance of KOMIC IDS and the best obtained accuracy results was 97.14%, precision 95.53%, recall 99.46%, and F1-score 97.46% using UNSW-NB15 dataset. However, the work used only one dataset, unable to detect the type of attack, and the obtained accuracy can be improved further [32].

Hameed Lafta Saad (2024) proposed a hybrid approach for network intrusion detection based on cuckoo algorithm and perceptron neural network. The proposed approach enhances the accuracy of the existing IDs by 1%. Cuckoo algorithm employed for selecting subset of the features from the IDs dataset based on number of characteristics and the perceptron neural network applied for features attributes analysis. The proposed approach evaluated using KDD-CUP99 dataset and achieve a detection accuracy 89.8%, precision 93.41%, recall 99.13%, and F1-score 97.7%. However, the work used only one IDS dataset, and the obtained accuracy can be improved further [33].

Muhammad A. et al. (2024) presented a method for selecting features for IDS based on thresholding. The proposed method combines three machine learning methods which are mutual information, thresholding feature selection, and XGBoost classifier. The dependency between the input features and the target features of the IDS dataset is measured using the mutual information method. In thresholding, the optimal number of the selected features is determined for better classification results. Besides that, XGBoost applied to classify the selected features. The method evaluated using three different IDS datasets and achieved an accuracy 87.63%, 80.51, and 99.89% for UNSW-NB15, NSL-KDD, and CIC-IDS2017 datasets respectively. However, computational training time, and the obtained results can be improved further [34].

Anselme R. et al. (23) proposed an attacks detection approach in wireless sensor network (WSN). The proposed approach uses Stochastic Machine Learning (SML) based on Hidden Markov Models (HMMs) and Gaussian Mixture Models (GMMs). Principal Component Analysis (PCA) applied for reduction of the WSN dataset dimension. HMMs and GMMs trained using Expectation-Maximization iterative machine learning for better classification performance. The achieved accuracy by using HMMs and GMMs was 94.55% using WSN-DS dataset. However, the work used only one dataset and the obtained accuracy still needs further improvement [35].

Md. Alamin. et al. (2024) presented an approach that integrates machine learning (ML) techniques with the Synthetic Minority Oversampling Technique Tomek Link (SMOTE-TomekLink) algorithm to innovative intrusion detection. The approach enhanced the intrusion detection accuracy in WSN dataset by balancing the input data in the dataset, eliminates Tomek links, and collects minority cases. The approach achieves an accuracy 99.92% using WSN-DS dataset. The significant limitation is the computational cost associated with resampling and feature scaling processes, as the authors did not utilize feature selection to reduce computational complexity. Additionally, the model's performance may be influenced by the choice of hyper parameters and the quality of the training data [36].

All the above mentioned existing works still suffer from the diversity of attacks in various IDS datasets and the obtained results cand be improved further. Although, several works improved the obtained results for solving the IDS via various

enhancement, the problems still need an optimal and accurate approach. This open research is, inspired us for propose the CNPO to overcame the gaps in other feature selection methods.

3. The proposed approach

The proposed intrusion detection work considers many stages. These stages collaborate to obtain the best possible detection results. These stages include: Data collection, pre-processing, feature selection, classification, and evaluation. The main stages of the proposed work illustrated in Fig. 1.

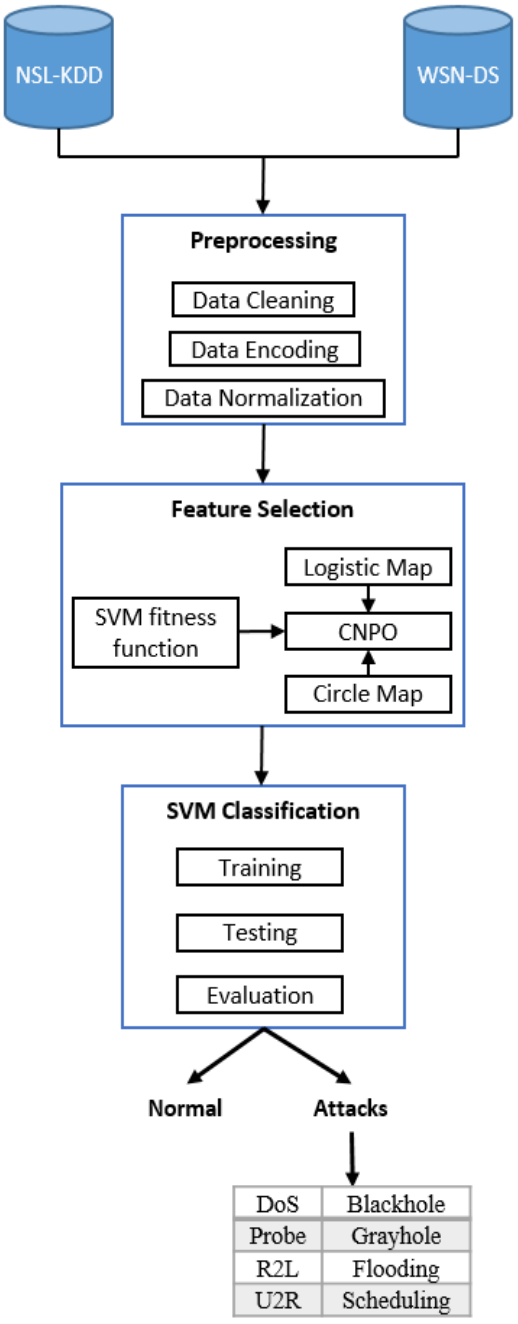


Figure. 1 Proposed Approach

3.1 Data collection

In the proposed work, two datasets are used in order to evaluate the system performance. The first dataset is NSL-KDD dataset [38], and the second one is WSN-DS [39]. NSL-KDD dataset is updated version of KDD'99 dataset.

The data set has many records and each record has 42 attributes (39 numerical and 3 symbolic). 41 attributes represent data characteristic and the last one class label which represents the attack type. The label includes the normal data and four types of attacks which are (denial of service DoS, Probe, R2L Remote to user, U2R User to Root). The dataset has two files one for train called (KDDTrain +) with 125973 records and the another one for testing called (KDDTest +) with 22544 records. Table 1 shows the number of instances in NSL-KDD dataset.

WSN-DS dataset is built for Dos attack detection. The data of WSN-DS collected using LEACH routing protocol. The dataset has 374661 records which represent the normal and four types of attacks. These attacks include: Blackhole, Grayhole, Flooding, and Scheduling. The details of the WSN-DS dataset are presented in Table 2.

Table 1. Details of NSL-KDD dataset

Attack	Instances	
	Training set	Testing set
Normal	67343	9711
DoS	45927	7456
Probe	11656	2421
R2L	995	2756
U2R	52	200
Total	125973	22544

Table 2. Details of WSN-DS dataset

Attack	Instances	
	Training set	Testing set
Normal	204174	135892
Blackhole	5999	4050
Grayhole	8653	5943
Flooding	1963	1349
Scheduling	7004	2631
Sum	224796	149865

3.2 Data pre-processing

In the pre-processing stage the NSL-KDD and WSN-DS datasets go through three crucial steps which are data cleaning, encoding and Normalization. The cleaning process removes missing and unnecessary values, such as the "num_outbound_cmds" values in NSL-KDD, which are always 0. As a result, the attribute values become beneficial for the proposed system's next stages.

Since the datasets have some attributes with text representation (like the attack types, protocols, etc.), and the next stages work only with numerical values, an encoding process is applied to convert the text to numerical representation using indexing in the second step of the pre-processing stage. Furthermore, each attack type will be labelled, as will normal behaviour, for use in the classification stage.

The last step of the pre-processing is data normalization. The normalization step reduces the effect of variance of the numerical data range on the classification stage. In the proposed work, the data is normalized into scale [0,1] using Eq. (1) [40].

$$X_{Scaled} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

3.3 Feature selection

One of the most important stage in the proposed work is the feature selection. An efficient feature section stage leads for better detection results. In the proposed work an improved version of NPO called CNPO for feature selection process using two chaotic maps (logistic and circle) and proposed fitness function (based on SVM) is proposed. The CNPO select the relevant features with highest impact on the obtained results based on optimal features criteria. The proposed feature selection method is shown in Fig. 2.

3.3.1. Nomadic people optimizer (NPO) [27]

NPO is a metaheuristic algorithm that inspired by the life style of the nomadic communities in the desert searching for the useful life sources to move toward it. NPO designed based on the multi-swarm approach with several clans for finding the best solution via following the leader position.

1- NPO Terminology

The algorithm has several main parameters that describe the terminologies of the NPO:

- Leader (σ): the best local current solution.
- Best Leader (σ^E): the best global current solution (used in the periodical meeting).

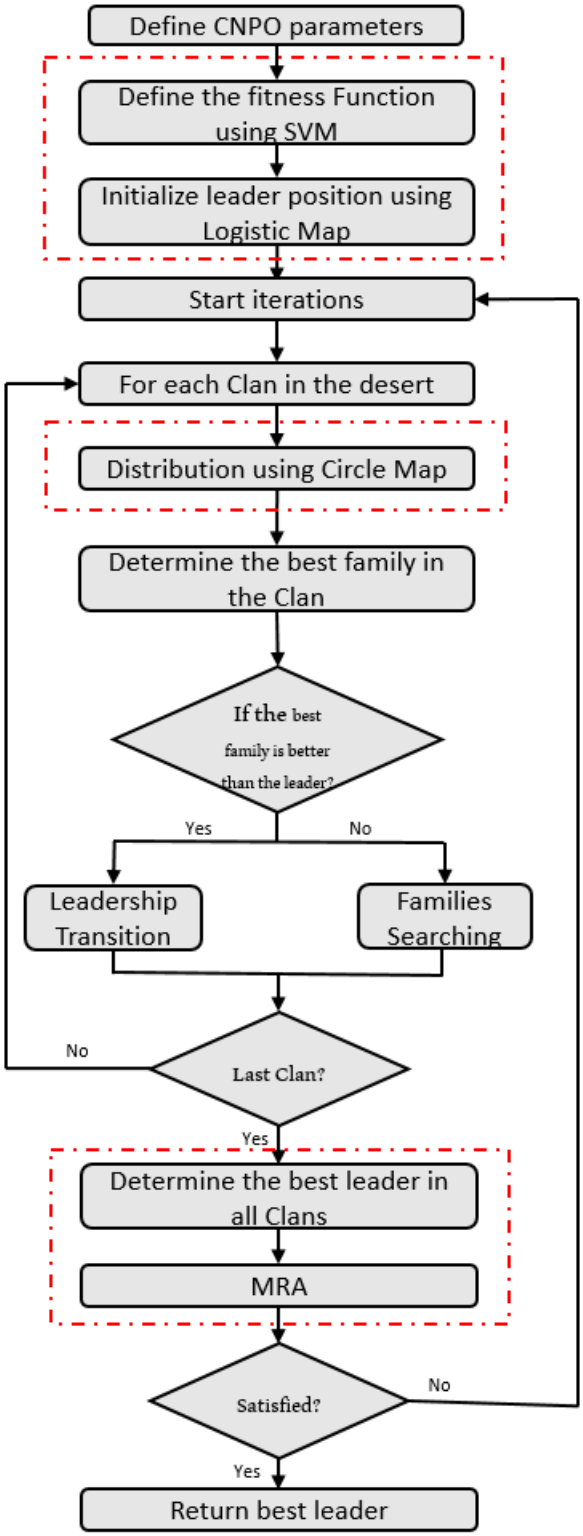


Figure. 2 Proposed Feature Selection Approach (CNPO)

- Normal Leader (σ^N): the other leaders (except the best one).
- Family (x): the clan member with lower fitness than leader.
- Clan (c): leaders with their families (one leader for each family).

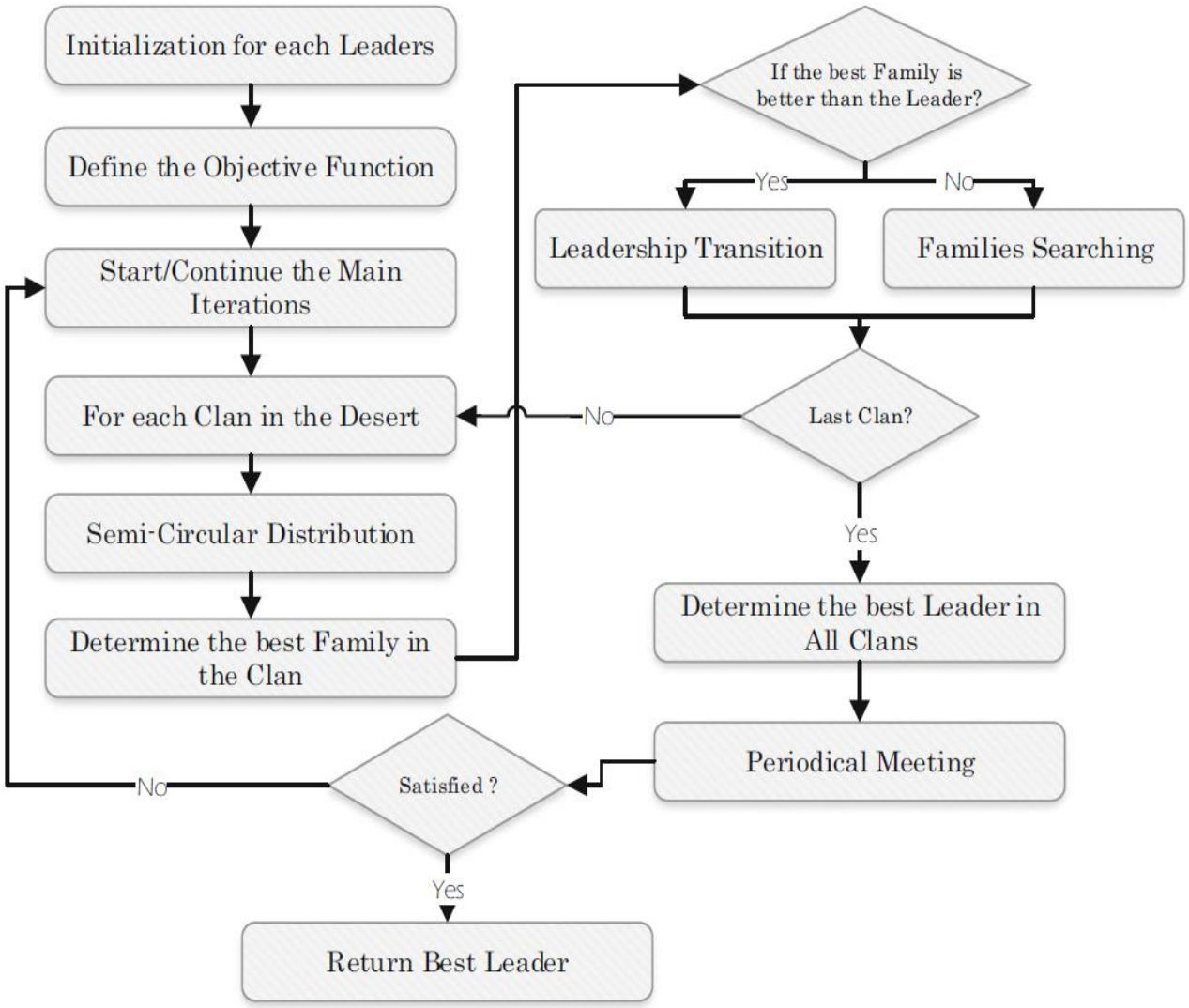


Figure. 3 NPO Algorithm

- Fitness function ($f(x)$): the goodness of position evaluation.
- Direction (Ψ): guide σ^N into σ^E.

2- NPO Algorithm

The NPO algorithm contents five main steps which are: initialization, semi-circular distribution, searching families, transition of leadership, and periodical meeting. Fig. 3 shows NPO flowchart within main steps of the algorithm.

Step 1: Initialization

Initialize the leaders (σ) randomly, $\sigma_i = \{\sigma_1, \sigma_2, \dots, \#c\}$using Eq. (2):

$$\overrightarrow{\sigma_c} = (UB - LB) \times R + LB \quad (2)$$

Step 2: Semi-circular distribution (Exploitation - local search)

The distribution of the families (x), where $X_i = \{X_1, X_2, \dots, \#x\}$ around the leader σ. The distribution equation illustrated below:

$$X = \left(Rd \times \sqrt{R_1}\right) \times \cos(\theta) + X_o \quad (3)$$

$$Y = \left(Rd \times \sqrt{R_2}\right) \times \sin(\theta) + Y_o \quad (4)$$

In scenario of finding the family position, Eq. (5) is used to find the X coordinate:

$$\overrightarrow{x_c} = \overrightarrow{\sigma_c} \times \sqrt{R} \times \cos(\theta) \quad (5)$$

Step 3: Searching for families (Exploration - global search)

When there is no new best local solution in NPO, the global search is considered. The families start to search randomly far from the current best local

solution in various directions. The random steps are based on the Lévy flight as shown in Eq. (6):

$$\overrightarrow{X_i^{new}} = \overrightarrow{X_i^{old}} + (a_c \times \left(\overrightarrow{\sigma_c} - \overrightarrow{X_i^{old}}\right) \oplus \text{Lévy}) \quad (6)$$

The area of clan (a_c) can calculated using Eq. (7):

$$a_c = \frac{\sum_{i=1}^{\Phi} \sqrt{(\overrightarrow{\sigma_c} - \overrightarrow{X_i^{old}})^2}}{\Phi} \quad (7)$$

The value of a_c has a great impact on the search process. When its low, the distribution will be in small circle. However, when a_c value is high, the distribution will be in large space (far from current a_c). The families search use Lévy flight for moving to another space in different directions. Lévy flight (λ_c) formula is illustrated in Eq. (8).

$$\text{Lévy} \backsim \text{u} = t^{-\lambda} \ (1 \leq \lambda \leq 3) \quad (8)$$

Step 4: Transition of Leadership (Exploitation)

The transition of the leadership is done when there is there is a better fitness value of the new family in clan better than the fitness value of the current leader, the family becomes the leader of the clan.

Step 5: Meeting Room Approach (MRA)

Periodical meeting is similar to the initialization except the leader's distribution. The meeting occurs between the leaders (normal leaders), and the best leader (with best solution) guide the other leader to find the better locations with best solutions. The position variance between leader position calculated using Eq. (9):

$$\Delta Pos = \Psi \left(\frac{\sqrt{\sum_i^D (\sigma^E - \sigma_c^N)^2}}{\#D}\right) \quad (9)$$

For guiding the normal leaders to better position, the following formula is used to find the direction variable Ψ:

$$\Psi = \begin{cases} 1 & f(f(\sigma^E) \geq 0 \\ -1 & Otherwise \end{cases} \quad (10)$$

After that, the normal leaders update their position via Eq. (11):

$$\overrightarrow{\sigma_c^{new}} = \overrightarrow{\sigma_c^N} + \Delta Pos(\sigma^E - \sigma_c^N) \times \frac{IT}{\#T} \quad (11)$$

The position of all normal leaders are updated in MRA. In case the leader has better new position than the old one, the leader will remain in its current position. The pseudo code on NPO is illustrated in Alg. (1).

Alg. (1)

1. Determine the NPO parameters: Clans(#Clans), Families (Φ), Iterations(#T)
2. Define the fitness function $f(x)$
3. Initialize the leaders σ_c^o = {1, 2, 3, … , #Clans}
4. Find the fitness value for each leader using $f(x)$
5. **Repeat** (Itr.)
6. **For** c = 1 to #Clans
7. Apply semi-circular distribution using Eq. 5
8. Find the fitness value for each solution x_i^c using $f(x)$
9. Set the best x_i^c in the *c* as σ_c^B
10. **IF** σ_c^B better than σ_c^o then $\sigma_c^o = \sigma_c^B$
11. **Else** Explore the search space:
12. Find the avg. distance between all families using Eq. 7
13. Move family into the new position using Eq. 6
14. Find the fitness value for each solution x_i^c using $f(x)$
15. Set the best x_i^c in the *c* as σ_c^B
16. **IF** σ_c^B better than σ_c^o then $\sigma_c^o = \sigma_c^B$
17. **End IF**
18. **End For**
19. Apply MRA
20. Loop Until (Itr. > #T)
21. Return the best leader σ^E

3.3.2. Chaotic maps [41]

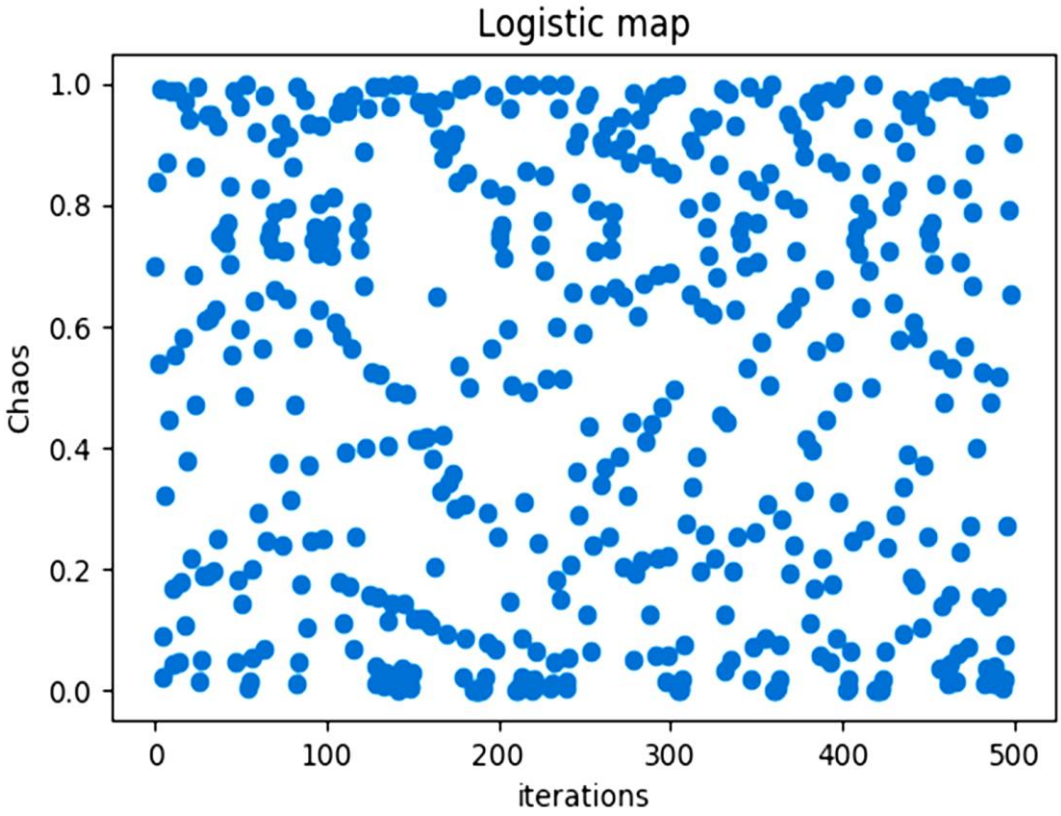
Chaotic maps are an evolution functions that generate arbitrary pattern through exhibit the chaotic behaviour. The chaotic parameters may perform in discrete-time or in continues-time. Chaotic maps are also a kind of pseudo randomness, however, since they are derived from the chaotic system, most of them would be bounded, irregular and sensitive to the initial conditions. The most common chaotic maps that successfully applied in several systems are shown in Fig. 4. In this work Logistic and circle maps are used.

1- Circle Map

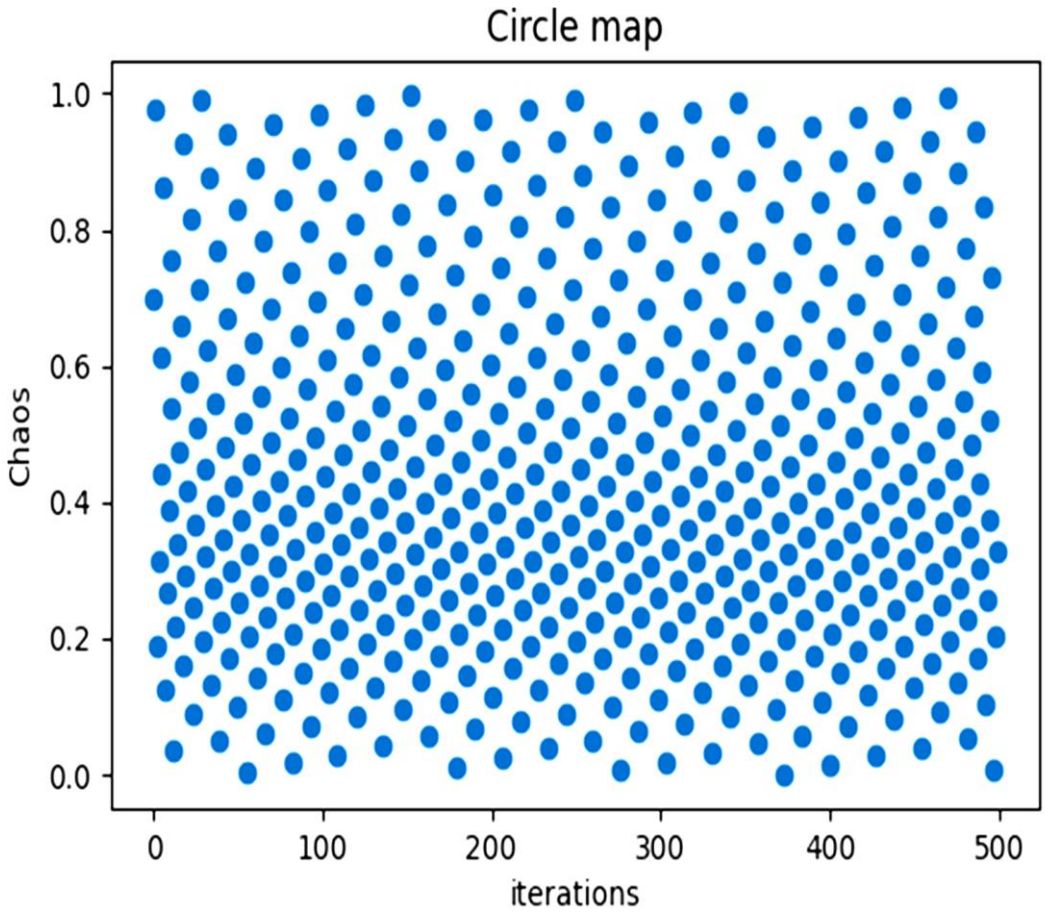
The circle map is represented by the sine function as in Eq. (12) [42]:

$$x_{k+1} = (x_k + b - \frac{2}{2\pi}\sin(2\pi x_k))mod(1) \quad (12)$$

When a = 0.5 and b = 2, the circle map generates chaos sequence in (0-1). The mod (1) represents the reminder of the division by 1.



(a)



(b)

Figure. 4 Chaotic Maps: (a) Logistic and (b) Circle

2- Logistic Map

The logistic map is a classical chaotic map from the nonlinear dynamic biological population evidencing the chaotic behaviour [42].

$$x_{k+1} = ax_k(1 - x_k) \quad (13)$$

The x_0 initially = {0, 0.25, 0.5, 0.75, 1} and the map generate chaos within values in [0,1] interval.

3.3.3. Support vector machine (SVM) [43]

SVM is a supervise machine learning model that used for classification, regression, and outlier's detection. SVM separate data into classes via finding the optimal hyperplane between the data. Thus, maximize the distance between the hyperplane (margin) and the support vectors (the closet points from class). The hyperplane presents in Eq. (14):

$$w^T.x + b = 0 \quad (14)$$

The prime form of SVM shown in Eq. (15):

$$minimize\ \frac{1}{2}\ \|w\|^2 \quad (15)$$

Where

$$y_i(w^T.x_i + b) \geq 1,\ \forall_i = 1, \dots, n \quad (16)$$

For high dimensional problem, the SVM solve it via Eq. (17):

$$maximize\ \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j y_i y_j (x_i x_j) \quad (17)$$

Where

$$\sum_{i=1}^{n} \alpha_i y_i = 0 \quad and \quad 0 \leq \alpha_i \leq C, \quad \forall_i = 1, \dots, n \quad (18)$$

However, the data may not linearly separable in the feature space. In this case, SVM perform kernel function to handle this issue. The kernel function map the original data into a high dimension space to easily find the hyperplane. The common SVM kernel functions are: linear kernel, polynomial kernel, radial basis function (RBF), and sigmoid kernel. Choosing the right kernel in SVM depend on several factors such as: data, complexity, model type, and parameters tuning.

3.3.4. Proposed feature selection algorithm

In this paper an improved version of NPO based on chaotic map, and SVM called CNPO is proposed. The proposed CNPO employed for feature selection task. CNPO enhance the classification accuracy by selecting the optimal relevant features. The inspiration of improving the standard NPO start from initializing the leaders using logistic map. This step creates a high diversity for leaders using Eq. (13) and enhance the performance and results of standard NPO.

In the second step of CNPO a proposed fitness function based on SVM is presented. The proposed fitness function evaluates all the generated solution using the following formula:

$$min\ f(B_i) = a \times SVM(D) + (1 - a) \times \frac{selected\ features}{all\ features} \quad (19)$$

Minimize this formula will lead to minimum error rate for the selected features (solutions).

In the next step of NPO the local search (exploitation) is enhanced using circle map. The families are distributed around the leader using Eq. (12). This step makes the distribution of the families much better around the corresponding leader than the original step in Eqs (3) and (4).

In the periodical meetings the updating Eq. (11) of the normal leaders is modified by adding random number (CR) to enhance the exploration. Eq. (20) shows the normal leader position update using CNPO.

$$\overrightarrow{\sigma_c^{new}} = \overrightarrow{\sigma_c^N} + \Delta Pos(\sigma^E - \sigma_c^N) \times CR \times \frac{IT}{\#T} \quad (20)$$

The overall steps of the proposed CNPO illustrated in Alg. (2).

Alg. (2)

1. Determine the NPO parameters: Clans(#Clans), Families (Φ), Iterations(#T)
2. Define the fitness function $f(x)$ via Eq. 19
3. Initialize the leaders σ_c^o = {1, 2, 3, … , #Clans} using logistic map Equation 13
4. Find the fitness value for each leader using Eq. 19
5. **Repeat** (Itr.)
6. **For** c = 1 to #Clans
7. Apply circle map distribution using Equation 18
8. Find the fitness value for each solution x_i^c using Eq. 19
9. Set the best x_i^c in the *c* as σ_c^B
10. **IF** σ_c^B better than σ_c^o then $\sigma_c^o = \sigma_c^B$
11. **Else** Explore the search space:

12. Find the avg. distance between all families using Eq. 7
13. Move family into the new position using Eq. 6
14. Find the fitness value for each solution x_i^c Eq. 19
15. Set the best x_i^c in the c as σ_c^B
16. **IF** σ_c^B better than σ_c^o then $\sigma_c^o = \sigma_c^B$
17. **End IF**
18. **End For**
19. Apply MRA using Eq. 20
20. Loop Until (Itr. > #T)
21. Return the best leader σ^E

3.4 Classification

In the classification stage SVM is used to classify the selected features from previous stage based on features class (label). The input data is separated into training and testing group using cross validation approach. 70% of the data selected for training and 30% is used for testing. The results of classification will be a class label of the input features either (normal data or attack types). In the proposed work RBF kernel is used in SVM for classification.

3.5 Evaluation measurements

There are different evaluation measurements that used to evaluate the performance of the proposed work. In this paper several metrics are used which are: accuracy, precision, recall, F1-score, and confusion matrix (TP, TN, FP, and FN) as shown in the following Equations [44].

$$Accuracy \frac{TP+TN}{TP+TN+FP+FN} \times 100 \quad (21)$$

$$\text{Precision} = \frac{TP}{TP+FP} \times 100 \quad (22)$$

$$\text{Recall} = \frac{TP}{TP+FN} \times 100 \quad (23)$$

$$F-measure = 2 \times \frac{Precision \times Recall}{Precision+Recall} \times 100 \quad (24)$$

4. Results

The proposed work tested on two intrusion detection datasets which are NSL-KDD, and WSN-DS with Windows 11 environment and Python programming language. 70% of the datasets used for training and 30% used for testing the proposed work. Besides, Eqs. (21)-(24) are used for evaluate the work performance. The results of applying the proposed work illustrated in Table 3.

Table 3. Result of NPO and CNPO algorithms

Meth.	Dataset	Acc. %	Prec. %	Rec. %	F1%
NPO	NSL-KDD	93.5	94.4	94.4	96.6
	WSN-DS	92.3	94.1	94.1	96
CNPO	NSL-KDD	99.96	99.6	99.6	99.9
	WSN-DS	99.98	99.98	99.98	99.9

Table 4. Results of Different Classifiers

Class.	Dataset	Acc. %	Prec. %	Rec. %	F1%
KNN	NSL-KDD	95.2	95.2	95.2	97.5
	WSN-DS	96	96	96	97.9
ANN	NSL-KDD	97.9	97.9	97.9	98.9
	WSN-DS	98.5	98.5	98.5	99.2
SVM	NSL-KDD	99.96	99.96	99.96	99.9
	WSN-DS	99.98	99.98	99.98	99.9

Table 5. Results of SVM Kernels

Class.	Ker.	Data.	Acc. %	Prec. %	Rec. %	F1%
SVM	Linear	NSL-KDD	98.2	98.2	98.2	99
		WSN-DS	98.5	98.5	98.5	99
	Sigmoid	NSL-KDD	98	98	98	98
		WSN-DS	98.2	98.2	98.2	99
	Polynomial	NSL-KDD	99.2	99.2	99.2	99.1
		WSN-DS	99.5	99.5	99.5	99.7
	RBF	NSL-KDD	99.96	99.96	99.96	99.9
		WSN-DS	99.98	99.98	99.98	99.9

The proposed CNPO outperform the standard NPO in the two used datasets. CNPO achieve accuracy 99.96% and 99.98% for NSL-KDD, and WSN-DS respectively, which is better about 6% than the NPO results. Besides, it achieved precision 99.6%, 99.98%, recall 99.6%. 99.98, and F-score 100 for both datasets NSL-KDD, and WSN-DS. Another comparison is made between the SVM classifier and

Table 6. Results of Proposed Approach and Other Works

Ref.	Dataset	Method	Accuracy	Precision	Recall	F-score
[31] 2023	NSL-KDD	GBA	96.96%	96.96%	96.96%	98.4%
[30] 2023	NSL-KDD	PACENIDS PACENIDS_ IFFS	94.47% 97.67%	89.41% 96.30%	89.22% 96.76%	89.31% 96.53%
[37] 2023	WSN-DS	GNB+SGD	98%	96%	96%	97%
[32] 2024	UNSW-NB15	KOMIG IDS	97.14%	95.53%	99.46%	97.46%
[33] 2024	KDD-CUP99	Cuckoo Algorithm	89.8%	93.41%	99.13%	97.7%
[29] 2024	NSL-KDD	SVM TD RF	95.2% 92.7% 94.5%	92.6% 89.8% 91.7%	94.3% 91.5% 93.2%	93.4% 90.6% 92.4%
[34] 2024	UNSW-NB15 NSL-KDD CIC-IDS2017	XGBoost + Mutual Information+ Thresholding	87.63% 80.51% 99.89%	96.35% 68.06% 99.75%	83.66% 96.73% 99.60%	89.56% 79.51% 99.68%
[35] 2024	WSN-DS	HMMs+GMMs	94.55%	-	-	-
[36] 2024	WSN-DS	SMOTE-Tomek	99.92%	99.92%	99.92%	99.92%
Proposed	NSL-KDD WSN-DS KDD-CUP99 CIC-IDS2017 UNSW-NB15	CNPO + SVM	99.96% 99.98% 92.23% 99.92% 89.99%	99.96% 99.98% 92.23% 99.92% 89.99%	99.96% 99.98% 92.23% 99.92% 89.99%	99.9% 99.9% 95.9% 99.9% 94.6%

other machine learning classifiers with the proposed CNPO as shown in Table 4.

In Table 4, using SVM with the proposed CNPO achieved the highest accuracy, precision, recall, and f-score results than other classifiers. Another comparison is performed using the SVM kernels to obtain a better classification result. Experimental shows an outperform of RBF kernel comparing to other SVM kernels in both datasets as shown in Table 5.

The proposed approach is compared with other recent works that have been done for intrusion detection using various datasets and method as shown in Table 6.

5. Performance analysis

In depth discussion and analysing the results of the proposed intrusion detection system based on CNPO is presented. CNPO superior from the standard NPO standard NPO in selecting the most relevant features that leads for better detection results using NSL-KDD, and WSN-DS datasets.

In another hand, SVM outperform KNN and ANN in the classification accuracy with RBF kernel. The SVM classifier archived the highest classification accuracy in both datasets NSL-KDD, and WSN-DS using RBF kernels within parameters gamma = 0.09, and c = 1.0.

Furthermore, the proposed approach outperforms several recent intrusion detection work as in Table 6 using several IDS datasets in terms of accuracy, precision, recall, and f-score. The proposed approach outperforms the works that use WSN-DS datasets by 1.98% for the work that used GNB and SGD. Besides, it outperforms the works that used SVM, TD, and RF in range 3%-7%. The best accuracy 99.92% of the existing works applied on WSN-DS dataset was using SMOTE-Tomek and the proposed approach still get the leads by 0.06% of the obtained accuracy. In another hands, the proposed approach outperforms the works that using NSL-KDD by 0.1%-4% of the

obtained accuracy using various machine learning methods. Furthermore, the proposed approach outperforms the works that using other IDS datasets UNSW-NB15, CIC-IDS2017, and KDD-CUP99 and still got the better obtained results. In addition, the proposed work tested on KDD-CUP99, CIC-IDS2017, and UNSW-NB15 datasets and achieved an accuracy 92.23%, 99.92%, and 89.99% respectively which are better than the obtained results by other recent works [32-34].

6. Conclusion

This paper proposed an accurate machine learning approach for intrusion detection in networks and wireless sensor networks using chaotic maps, NPO, and SVM classifier. The proposed approach has several stages which are: Data collection, pre-processing, feature selection, classification, and evaluation. The experiments used two open source IDS datasets NSL-KDD and WSN-DS.

The proposed approach achieved a high detection results using the proposed CNPO feature selector, and SVM classifier. The best obtained results for NSL-KDD dataset were accuracy 99.96%, precision 99.96%, recall 99.96%, F1-score 99.9%. Also, the obtained results for WSN-DS dataset were accuracy 99.98%, precision 99.98%, recall 99.98%, F1-score 99.9%. In addition, the proposed approach compared with the most recent works for the IDS and outperform the accuracy of the existing works.

Finally, future studies can focus at other approaches to improve the NPO exploration and exploitation processes by enhance the initial population in order to reach an optimal compromise.

Notation List

Notation	Meaning
X	Original value in dataset
X_{Scaled}	The normalized value
UB	Upper bound
LB	Lower bound
Rand	Random value between [0,1]
$\overrightarrow{\sigma_c}$	The leader position of clan
X_o, Y_o	The coordinates of origin point in circle
R_1, R_2	Random coordinates of a point in circle
θ	Angle value of the point in circle
$\overrightarrow{X_c}$	The family position
R	Random value between [0,1]
$\overrightarrow{X_i^{new}}$	New position of the family
$\overrightarrow{X_i^{old}}$	Old position of the family
a_c	The clan area
Φ	Number of families in each clan
x_c^i	The normal families
λ_c	Lévy flight
σ^E	Best leader position
σ_c^N	Normal leaders position
$\#D$	Number of dimensions
ΔPos	The normalized distance
$\overrightarrow{\sigma_c^{new}}$	New position of the normal leader
IT	Current iteration
$\#T$	Total number of iterations
$mod\ (1)$	The reminder after dividing by 1
x_n	Current chaos
x_{n+1}	Chaos at next iteration
W	Normal vector
b	Offsite distance
α	The Lagrange multiplier
a, CR	Random number between [0,1]
$f(B_i)$	Fitness function
TP	True positive
TN	True negative
FP	False positive
FN	False negative

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

The contribution for each author in this research is as follows: First and second are responsible for conceptualization, methodology, writing original draft preparation, writing review and editing, visualization. Third author is responsible for data curation, formal analysis, investigation, and validation.

Acknowledgments

We gratefully acknowledge the Lebanese University and Imam Ja'afar Al-Sadiq University for providing research facilities and resources that facilitated this study. Their assistance was instrumental in the completion of this research project.

References

[1] S. Mohamed, and R. Ejbali, "Deep SARSA-based reinforcement learning approach for anomaly network intrusion detection system", *International Journal of Information Security*, Vol. 22, pp. 235-247, 2023, doi: 10.1007/s10207-022-00634-2

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

118745, 2023, doi: 10.1016/j.eswa.2022.118745

[3] H. Yang, J. Xu, Y. Xiao, and L. Hu, "SPE-ACGAN: A Resampling Approach for Class Imbalance Problem in Network Intrusion Detection Systems", *Electronic*s, Vol. 12, No. 15, pp. 3323, 2023.

[4] A. Singh, P.K. Chouhan, and G.S. Aujla, "SecureFlow: Knowledge and data-driven ensemble for intrusion detection and dynamic rule configuration in software-defined IoT environment", *Ad Hoc Networks*, Vol. 156, pp. 103404, 2024.

[5] N. Jeffrey, Q. Tan, and J. R. Villar, "A hybrid methodology for anomaly detection in Cyber-Physical Systems", *Neurocomputing*, Vol. 568, pp. 127068, 2024.

[6] J. Azimjonov, and T. Kim, "Stochastic gradient descent classifier-based lightweight intrusion detection systems using the efficient feature subsets of datasets", *Expert Systems with Applications*, Vol. 237, pp. 121493, 2024.

[7] H. M. Saleh, H. Marouane, and A. Fakhfakh, "Stochastic Gradient Descent Intrusions Detection for Wireless Sensor Network Attack Detection System Using Machine Learning", *IEEE Access*, Vol. 12, pp. 3825-3836, 2024.

[8] E. Osa, P. E. Orukpe, and U. Iruansi, "Design and implementation of a deep neural network approach for intrusion detection systems", *e-Prime-Advances in Electrical Engineering, Electronics and Energy*, Vol. 7, pp. 100434, 2024.

[9] K. Cengiz, S. Lipsa, R. K. Dash, N. Ivković, and M. Konecki, "A novel intrusion detection system based on artificial neural network and genetic algorithm with a new dimensionality reduction technique for UAV communication", *IEEE Access*, Vol. 12, pp. 4925-4937, 2024.

[10] L. D. Manocchio, S. Layeghy, W. W. Lo, G. K. Kulatilleke, M. Sarhan, and M. Portmann, "Flowtransformer: A transformer framework for flow-based network intrusion detection systems", *Expert Systems with Applications*, Vol. 241, p. 122564, 2024.

[11] S. Sridevi, R. Prabha, K. N. Reddy, K. M. Monica, G. A. Senthil, and M. Razmah, "Network Intrusion Detection System using Supervised Learning based Voting Classifier", In: *Proc. of 2022 International Conference on Communication, Computing and Internet of Things (IC3IoT)*, pp. 1-6, 2022, doi: 10.1109/IC3IOT53935.2022.9767903.

[12] H. M. Farghaly and T. A. El-Hafeez, "A high-quality feature selection method based on frequent and correlated items for text classification", *Soft computing*, Vol. 27, No. 16, pp. 11259-11274, 2023, doi: 10.1007/s00500-023-08587-x.

[13] F. Macedo, R. Valadas, E. Carrasquinha, M. R. Oliveira, and A. Pacheco, "Feature selection using Decomposed Mutual Information Maximization", *Neurocomputing*, Vol. 513, pp. 215-232, 2022, doi: 10.1016/j.neucom.2022.09.101.

[14] S. Rosidin, Muljono, G. F. Shidik, A. Z. Fanani, F. Al Zami, and Purwanto, "Improvement with Chi Square Selection Feature using Supervised Machine Learning Approach on Covid-19 Data", In: *Proc. of International Seminar on Application for Technology of Information and Communication (iSemantic)*, pp. 32-36, 2021, doi: 10.1109/iSemantic52711.2021.9573196.

[15] N. Elssied, A. Prof. Dr. O. Ibrahim, and A. Hamza Osman, "A Novel Feature Selection Based on One-Way ANOVA F-Test for E-Mail Spam Classification", *Research Journal of Applied Sciences, Engineering and Technology*, Vol. 7, pp. 625-638, 2014, doi: 10.19026/rjaset.7.299.

[16] J. Cheng, J. Sun, K. Yao, M. Xu, and Y. Cao, "A variable selection method based on mutual information and variance inflation factor", *Spectrochim Acta A Mol Biomol Spectrosc*, Vol. 268, pp. 120652, 2022, doi: 10.1016/j.saa.2021.120652.

[17] N. Manju, B. S. Harish, and V. Prajwal, "Ensemble Feature Selection and Classification of Internet Traffic using XGBoost Classifier", *International Journal of Computer Network and Information Security*, Vol. 11, pp. 37-44, 2019, doi: 10.5815/ijcnis.2019.07.06.

[18] M. K. Alsmadi et al., "Intrusion Detection Using an Improved Cuckoo Search Optimization Algorithm", *J Wirel Mob Netw Ubiquitous Comput Dependable Appl*, Vol. 15, No. 2, pp. 73-93, 2022, doi: 10.58346/jowua.2024.i2.006.

[19] H. Lafta, "Network Intrusion Detection Using Optimal Perception with Cuckoo Algorithm", *Wasit Journal for Pure sciences*, Vol. 3, No. 1, pp. 95-105, 2024, doi: 10.31185/wjps.326.

[20] M. Ragab, S. M. Alshammari, and A. S. Al-Malaise Al-Ghamdi, "Modified Metaheuristics with Weighted Majority Voting Ensemble Deep Learning Model for Intrusion Detection System", *Computer Systems Science and Engineering*, Vol. 47, No. 2, pp. 2497-2512, 2023, doi: 10.32604/csse.2023.041446.

[21] M. Jeyaselvi et al., "A highly secured intrusion detection system for IoT using EXPSO-STFA

feature selection for LAANN to detect attacks", *Cluster Comput*, Vol. 26, No. 1, pp. 559-574, 2023, doi: 10.1007/s10586-022-03607-1.

[22] T. R. Ramesh, T. Jackulin, R. A. Kumar, K. Chanthirasekaran, and M. Bharathiraja, "Machine Learning-Based Intrusion Detection: A Comparative Analysis among Datasets and Innovative Feature Reduction for Enhanced Cybersecurity", *International Journal of Intelligent Systems and Applications in Engineering*, Vol. 12, No. 12s, pp. 200-206, 2024.

[23] B. Mohammed, and E. K. Gbashi," Intrusion Detection System for NSL-KDD Dataset Based on Deep Learning and Recursive Feature Elimination", *Engineering and Technology Journal*, Vol. 39, No. 07, pp. 1069-1079, 2021.

[24] H. Asgharzadeh, A. Ghaffari, M. Masdari, and F. S. Gharehchopogh, "An Intrusion Detection System on The Internet of Things Using Deep Learning and Multi-Objective Enhanced Gorilla Troops Optimizer.", *J Bionic Eng*, Vol. 9, pp. 1-27, 2024, doi: 10.1007/s42235-024-00575-7

[25] M. Hasanah, R. A. Putri, M. Aidie, R. Putra, and T. Ahmad, "Analysis of Weight-Based Voting Classifier for Intrusion Detection System", *International Journal of Intelligent Engineering and Systems*, Vol. 17, No. 2, pp. 190-200, 2024, doi: 10.22266/ijies2024.0430.17.

[26] S. Q. Salih and A. R. A. Alsewari, "A new algorithm for normal and large-scale optimization problems: Nomadic People Optimizer", *Neural Comput Appl*, Vol. 32, No. 14, pp. 10359-10386, 2020, doi: 10.1007/s00521-019-04575-1.

[27] S. T. Ahmed and S. M. Kadhem, "Optimizing Alzheimer's disease prediction using the nomadic people algorithm", *International Journal of Electrical and Computer Engineering*, Vol. 13, No. 2, pp. 2052-2067, 2023, doi: 10.11591/ijece.v13i2.

[28] A. Q. Mohammed, K. A. Al-Anbarri, and R. M. Hannun, "Introducing newly developed Nomadic People Optimizer (NPO) algorithm to find optimal sizing of a hybrid renewable energy", *IOP Conference Series: Materials Science and Engineering, IOP Publishing Ltd*, pp. 1-16, 2020, doi: 10.1088/1757-899X/928/2/022052.

[29] B. R. Maddireddy and B. R. Maddireddy, "A Comprehensive Analysis of Machine Learning Algorithms in Intrusion Detection Systems.", *Journal of Environmental Sciences and Technology (JEST)*, Vol. 3, No. 1, pp. 877-893, 2024.

[30] N. Girubagari and T. N. Ravi, "Parallel ABILSTM and CBIGRU Ensemble Network Intrusion Detection System", *International Journal of Intelligent Engineering and Systems*, Nol. 17, No. 1, pp. 93-107, 2024, doi: 10.22266/ijies2024.0229.10.

[31] S. S. Issa, S. Q. Salih, Y. D. Salman, and F. H. Taha, "An Efficient Hybrid Filter-Wrapper Feature Selection Approach for Network Intrusion Detection System", *International Journal of Intelligent Engineering and Systems*, Vol. 16, No. 6, pp. 261-273, 2023, doi: 10.22266/ijies2023.1231.22.

[32] A. S. Afolabi and O. A. Akinola, "Network Intrusion Detection Using Knapsack Optimization, Mutual Information Gain, and Machine Learning", *Journal of Electrical and Computer Engineering*, Vol. 2024, pp. 1-21, 2024, doi: 10.1155/2024/7302909.

[33] H. Lafta, "Network Intrusion Detection Using Optimal Perception with Cuckoo Algorithm", *Wasit Journal for Pure sciences*, Vol. 3, No. 1, pp. 95-105, 2024, doi: 10.31185/wjps.326.

[34] M. A. Faizin, D. T. Kurniasari, N. Elqolby, M. A. R. Putra, and T. Ahmad, "Optimizing Feature Selection Method in Intrusion Detection System Using Thresholding", *International Journal of Intelligent Engineering and Systems*, Vol. 17, No. 3, pp. 214-226, 2024, doi: 10.22266/ijies2024.0630.18.

[35] A. R. A. Moundounga, and H. Satori, "Stochastic machine learning based attacks detection system in wireless sensor networks", *J. Netw. Syst. Manag*. Vol. 32, No. 17, 2024.

[36] M. A. Talukder, S. Sharmin, M. A. Uddin, M. M. Islam, and S. Aryal, "MLSTL-WSN: machine learning-based intrusion detection using SMOTETomek in WSNs", *Int J Inf Secur*, Vol. 23, No. 3, pp. 2139-2158, 2024, doi: 10.1007/s10207-024-00833-z.

[37] H. M. Saleh, H. Marouane, and A. Fakhfakh, "Stochastic Gradient Descent Intrusions Detection for Wireless Sensor Network Attack Detection System Using Machine Learning", *IEEE Access*, Vol. 12, pp. 3825-3836, 2024, doi: 10.1109/ACCESS.2023.3349248.

[38] R. ZHAO, "NSL-KDD", *IEEE Dataport*, 2022, doi: https://dx.doi.org/10.21227/8rpg-qt98.

[39] J. Pan, Y. Zhuang, S. Fong, "the impact of data normalization on stock market prediction: using SVM and technical indicators", In: *Proc. of International Conference on Soft Computing in Data Science, Springer*, pp. 72-88, 2016.

[40] I. Almomani, B. Al-Kasasbeh, and M. Al-Akhras, "WSN-DS: A Dataset for Intrusion

Detection Systems in Wireless Sensor Networks", *J Sens*, Vol. 2016, 2016, doi: 10.1155/2016/4731953.

[41] Z. M. Gao, J. Zhao, and Y. J. Zhang, "Review of chaotic mapping enabled nature-inspired algorithms", *Math Biosci Eng.*, Vol. 19, No. 8, pp. 8215-8238, 2022, doi: 10.3934/mbe.2022383.

[42] A. H. Gandomi, X. S. Yang, "Chaotic bat algorithm", *J. Comput. Sci.*, Vol. 5, pp. 224-232, 2014, doi: 10.1016/j.jocs.2013.10.002.

[43] M. Hosseinzadeh, A. M. Rahmani, B. Vo, M. Bidaki, M. Masdari, and M. Zangakani, "Improving security using SVM-based anomaly detection: Issues and challenges", *Soft Comput*., Vol. 25, pp. 1-29, 2020, doi: 10.1007/s00500-020-05373-x.

[44] T. Saranya, S. Sridevi, C. Deisy, T. D. Chung, and M. K. A. A. Khan, "Performance analysis of machine learning algorithms in intrusion detection system: A review", *Procedia Computer Science*, Vol. 171, pp. 1251-1260, 2020.