frontiers | Frontiers in Big Data

TYPE Original Research
PUBLISHED 01 February 2023
DOI 10.3389/fdata.2023.1081466



OPEN ACCESS

EDITED BY
Namita Gupta,
Maharaja Agrasen Institute of Technology, India

REVIEWED BY
Sarita Gulia,
K.R. Mangalam University, India
Shweta Taneja,
Bhagwan Parshuram Institute of Technology, India

*CORRESPONDENCE
Richa Singh
✉ richa.singh081991@gmail.com

SPECIALTY SECTION
This article was submitted to Data Science, a section of the journal Frontiers in Big Data

RECEIVED 27 October 2022
ACCEPTED 12 January 2023
PUBLISHED 01 February 2023

CITATION
Singh R and Ujjwal RL (2023) Hybridized bio-inspired intrusion detection system for Internet of Things. *Front. Big Data* 6:1081466. doi: 10.3389/fdata.2023.1081466



# Hybridized bio-inspired intrusion detection system for Internet of Things

Richa Singh* and R. L. Ujjwal

University School of Information, Communication, and Technology, Guru Gobind Singh Indraprastha University, Dwarka, New Delhi, India

The Internet of Things (IoT) consists of several smart devices equipped with computing, sensing, and network capabilities, which enable them to collect and exchange heterogeneous data wirelessly. The increasing usage of IoT devices in daily activities increases the security needs of IoT systems. These IoT devices are an easy target for intruders to perform malicious activities and make the underlying network corrupt. Hence, this paper proposes a hybridized bio-inspired-based intrusion detection system (IDS) for the IoT framework. The hybridized sine-cosine algorithm (SCA) and salp swarm algorithm (SSA) determines the essential features of the network traffic. Selected features are passed to a machine learning (ML) classifier for the detection and classification of intrusive traffic. The IoT network intrusion dataset determines the performance of the proposed system in a python environment. The proposed hybridized system achieves maximum accuracy of 84.75% with minimum selected features i.e., 8 and takes minimum time of 96.42 s in detecting intrusion for the IoT network. The proposed system's effectiveness is shown by comparing it with other similar approaches for performing multiclass classification.

KEYWORDS

**Internet of Things, intrusion detection system, salp swarm algorithm, sine cosine algorithm, feature selection**

## 1. Introduction

IoT (Hussain et al., 2020) consist of several interconnecting devices known as things, with limited communication, computation, and storage capability. These devices can vary from simple household devices such as smart bulbs, smart fridges, smart meters, IP cameras, etc. to more complicated devices such as RFID, smart devices used in industry, heartbeat detectors, etc. IoT system (Al-Fuqaha et al., 2015) is gaining importance in every field including smart grid, smart agriculture, smart transportation, smart home, semantic web, etc. The exponential growth of IoT devices gives a platform to intruders to perform malicious activities. The intruder aims to exploit IoT resources by launching cyber-attacks against IoT devices and networks, which is destructive. An IoT system is susceptible to various types of threats (Chaabouni et al., 2019) such as man-in-middle attacks, denial of service, routing attacks, eavesdropping, etc. As a consequence, the IoT system security is of prime importance. An intrusion detection system (IDS) is one of the security methods for the IoT system, which aims to detect and report any breach. Based on the detection strategy (Khraisat and Alazab, 2021), the IDS is classified as signature IDS, anomaly IDS, and hybrid. In signature IDS (Thakkar and Lohiya, 2021), network traffic is examined for the attack signatures stored in the database, any match is considered as malicious. In anomaly IDS, the user profile is created based on analyzing the network usage. Any variance from the usage behavior is considered an anomaly. Hybrid IDS merges the advantages of signature IDS and anomaly IDS. The data produced by the IoT network is voluminous plus heterogeneous. Therefore, having an effective and efficient feature selection (FS) method is important. The FS method selects the best features from the original set based on certain criteria.

The FS methods are classified as, filter, wrapper, and hybrid (Balasaraswathi et al., 2017). In the filter method, various statistical measures such as correlation, information measure, and distance are used for selecting the feature subset. However, such methods don't interact with the classifier for the evaluation of feature subsets. The wrapper method interacts with the classifier for the evaluation of the selected feature subset. The hybrid method combines the merits of the filter and wrapper method.

Many works were published where metaheuristic algorithms (MHA) are used for the feature selection tasks of IDS. For instance, Kareem et al. (2022) proposed an efficient feature selection algorithm for IoT-based IDS. They hybridize the bird swarm algorithm (BSA) with the gorilla troops optimization (GTO) algorithm for the feature selection task. The exploration part of gorilla troops optimization is modified using bird swarm algorithm. The proposed work provides better convergence results compared to other metaheuristic algorithms used for feature selection. An IDS for the IoT-based healthcare systems is proposed by Saif et al. (2022). The proposed work uses a genetic algorithm, differential evaluation, and partial swarm optimization (PSO) for selecting an optimal feature subset. Afterward, the classification of intrusive traffic is done using k-nearest neighbor, and a decision tree classifier. Haddadpajouh et al. (2020) proposed a IDS for the IoT edge layer. The feature selection is performed using the bio-inspired gray wolf optimization (GWO) algorithm. A multi-kernel support vector machine is used for the detection of malicious traffic. Another work by Mafarja et al. (2020) employed a modified whale optimization algorithm for reducing the dimension in IoT-based IDS. The whale optimization algorithm is modified using transfer functions, and its performance is better than the original whale optimization algorithm. Hosseini et al. (2022) proposed a hybridized bio-inspired algorithm for detection botnets in the IoT network. They hybridized salp swarm algorithm and slime mold algorithm for feature selection of network traffic. Alweshah et al. (2022) proposed an emperor penguin colony based feature selection approach for the IoT-based IDS. The classification of intrusive traffic is performed using k-nearest neighbor (KNN) classifier.

Work by Fatani et al. (2021) proposed an aquila optimizer (AO) based feature selection approach for detecting intrusion in the IoT framework. In this paper, feature extraction is performed by convolutional neural network, and afterward, aquila optimizer determines the best features. The proposed work is evaluated against four datasets and the result shows the efficiency of proposed work against other related work. Krishna and Arunkumar (2021) proposed a GWO and PSO-based IDS for an IoT environment. The random forest classifier performs multiclass classification using the optimal features selected from hybrid PSO and GWO. Experimental result shows the effectiveness of the proposed system against other similar work using the NSL-KDD and N-BaIoT dataset. Furthermore, work by Sarwar et al. (2022) proposed a feature selection algorithm for IDS. They detect optimal features using an improved dynamic sticky binary partial swarm optimization algorithm. Classification of IoT intrusive traffic is done using random forest (RF). Dahou et al. (2022) employed a reptile search algorithm for selecting optimal features in IoT framework. The proposed work is evaluated against multiple datasets such as NSL-KDD, BoT-IoT, KDDCup-99, and CICIDS-2017. Work by Priya et al. (2020) proposed a hybridized principal component analysis and gray wolf optimization based IDS for the Internet of Medical Things. The intrusive traffic is classified using deep neural network. The hybridized bio-inspired-based IDS is proposed by Davahli et al. (2020) for IoT system. They hybridized genetic algorithm and gray wolf optimization for optimal feature selection. Further, the intrusive traffic is classified using support vector machine classifier. The AWID dataset is used for the performance evaluation of this system. A smart IDS for IoT system is proposed by Keserwani et al. (2021). They employed hybridized GWO and PSO algorithm for feature selection task. The multiclass classification is performed using random forest classifier. The proposed work is evaluated against multiple datasets.

Hence, this paper proposes hybridized IDS for the IoT system. The feature selection is done by hybridizing the bio-inspired sine cosine algorithm (SCA), and salp swarm algorithm (SSA) for selecting the optimal feature subset. The proposed approach is compared with other bio-inspired algorithms used for the FS task in the IoT-based IDS. The dataset IoTID20 is used to verify and evaluate system performance. Multiclass classification is performed using two machine learning classifiers and afterward, their performance is analyzed. The performance of hybridized SCA-SSA with KNN and XGBoost classifiers is better compared to other metaheuristic algorithms used for feature selection. The paper is formulated as: Section 2 describes the material and methods used by the hybridized IoT-based IDS, and Section 3 describes the implementation result. Section 4 presents the discussion.

# 2. Material and methods

## 2.1. Background

### 2.1.1. Sine cosine algorithm (SCA)

This math-based algorithm is defined by Mirjalili (2016). SCA is motivated by trigonometric properties of sine and cosine functions for updating individual positions which provide an optimal solution for optimization problems. SCA is easy to implement, flexible and the probability of falling in local optima is low. However, SCA might suffer from premature convergence. The solutions are updated using the following equations:

$$X_{m,n}^{ite+1} = X_{m,n}^{ite} + rnd_1 \times \sin(rnd_2) \times \left| rnd_3 X_{Best\ n}^{ite} - X_{m,n}^{ite} \right| \quad if\ rnd_4 < 0.5 \tag{1}$$

$$X_{m,n}^{ite+1} = X_{m,n}^{ite} + rnd_1 \times \cos(rnd_2) \times \left| rnd_3 X_{Best\ n}^{ite} - X_{m,n}^{ite} \right| \quad if\ rnd_4 \geq 0.5 \tag{2}$$

where, $X_{m,n}^{ite}$ is a current solution of individual m in $n$-th dimension at iteration $ite$. $X_{Best\ n}^{ite}$ is the best solution at iteration $ite$ in the $n$-th dimension. $rnd_1$, $rnd_2, rnd_3, and\ rnd_4$ are random numbers. $rnd_2$ lies between$[0, 2\pi]$. $rnd_3$ controls the search agent mobility direction. $rnd_4$ is used to switch between two search methods. $rnd_1$ controls the exploration and exploitation phase and is updated using the following equation:

$$rnd_1 = k - \frac{k}{ite_{\max}} \times ite \tag{3}$$

$$rnd_2 = 2 \times \pi \times \ random\ number(0, 1) \tag{4}$$

where, k is constant, $ite_{\max}$ is the maximum iteration and *ite* is the current iteration.

### 2.1.2. Salp swarm algorithm (SSA)

This algorithm is proposed by Mirjalili et al. (2017). SSA is motivated by the hunting habits of salps in the sea. They belong to salpidae family, and their movement is alike jellyfish. They form a chain while living in a group. It required a few parameter settings and is simple to implement. However, SSA might suffer from premature convergence. The foremost salp in a chain is the leader, and others are followers. Leader position is determined with the following equation:

$$X_n^1 = FPos_j - crn_1 \left[ \left( UpBn - LoBn \right) crn_2 + LoBn \right] \ if\ crn_3 \ < \ 0.5 \tag{5}$$

$$X_n^1 = FPos_j + crn_1 \left[ \left( UpBn - LoBn \right) crn_2 + LoBn \right] \ if\ crn_3 \ \geq \ 0.5 \tag{6}$$

where, $X_n^1$ is the leader position, $FPos_j$ food position. *UpBn* is upper bound. *LoBn* is lower bound. $crn_2$ is a random number between [0,1] used to control the mobility step of the leader. $crn_3$ controls the switch between two position-updating equations. $crn_1$ is the control parameter that balances SSA execution. It is defined by the following equation:

$$crn_1 = 2e^{-\left(\frac{4\ ite}{ite_{\max}}\right)} \tag{7}$$

The follower position is determined using the following equation:

$$X_n^m = \frac{1}{2}\left(X_n^m + X_n^{m-1}\right) \tag{8}$$

where $X_n^m$ represents the *m*-th follower in *n*-th dimension.

## 2.2. Proposed system

A hybridized IDS for the IoT framework has been proposed in this section. Figure 1 depicts the proposed system and it is divided into the following parts including, data preparation, feature selection using SCA-SSA, classification, and detection, and performance evaluation.

### 2.2.1. IoTID20 dataset

The IoTID20 (Ullah and Mahmoud, 2020) dataset is used as collected data in this paper. The IoTID20 is the result of the testbed configuration of the smart home. Smart home devices such as laptops, tablets, Wi-Fi cameras, smartphones, and other devices are used to generate network flow data for the IoTID20 dataset. These devices are divided into categories of victim devices i.e., EZVIZ Wi-Fi camera, SKT NGU, and the attacking devices including smartphones, tablets, etc. all other IoT devices. This labeled dataset includes 83 features. The five attack category instances of the IoTID20 dataset used for performance evaluation are shown in Table 1.

### 2.2.2. Data preparation

The performance of the learning algorithm depends upon the type of data provided as input. Therefore, various data preparation techniques such as transformation, normalization, and sampling are employed to improve the quality of data. A vast amount of heterogeneous data is collected from IoT devices with numerous features. However, not all features are useful for the classification task, few of them are irrelevant and redundant. Therefore, these redundant and irrelevant attributes are eliminated. Afterward, categorical feature values are transformed into numerical values using the Label encoder function. The numerical feature value of the dataset might vary. Therefore, normalization is performed so that the values lie within a range of (0, 1).

In this paper, the standard scaler function is used for this purpose. The IoTID20 dataset is unbalanced. The instances of intrusive traffic are much more than instances of normal traffic. The unbalanced dataset degrades the performance of the classifier. Therefore, random sampling is used to make the IoTID20 dataset balanced.

### 2.2.3. Feature selection using hybrid SCA-SSA

The hybrid SCA-SSA that merges SCA with SSA is described in this section. The first part of SSA is enhanced using the SCA algorithm for updating the salp positions. During these modifications, the trigonometric functions of SCA are used to update the salp leader position. This enhances SSA flexibility for exploring search space and leads to providing optimal solutions. Afterward, the exploitation capability of salp is enhanced by adding the levy flight (LF) (Chawla and Duhan, 2018) to it. The follower position is updated using LF. LF controls the size of a step taken by salps. These enhancements lead to better convergence speed and avoid local optima. Therefore, the hybridization of SCA-SSA is shown in Algorithm 1 and is described as follows.

#### 2.2.3.1. First part

The first part of SSA is enhanced by using the trigonometric functions of SCA, and inertia weight, which improves the exploring capability of SSA. The leader position in SSA is updated the using following equation:

$$X_n^1 = w \times FPos_j + crn_1 \times \sin\ (rnd_2) \times \left| rnd_3\ X_{Best\ n}^{ite} - \ X_n^{ite} \right| \\ if\ crn_3 \ \geq \ 0.5 \tag{9}$$

$$X_n^1 = w \times FPos_j - crn_1 \times \cos\ (rnd_2) \times \left| rnd_3\ X_{Best\ n}^{ite} - \ X_n^{ite} \right| \\ if\ crn_3 \ < \ 0.5 \tag{10}$$

where, $w = 0.9$ is the inertia weight, $FPos_j$ food position in *j*-th dimension. $crn_1$ is obtained from Eq. (7). $rnd_2$ is obtained using Eq. (4). $rnd_3$ and $crn_3$ are random numbers. $X_{Best\ n}^{ite}$ is best solution at iteration *ite* in *n*-th dimension, and $X_n^{ite}$ is current solution of individual.

#### 2.2.3.2. Second part

The second part of SSA is enhanced by introducing levy flight function. Therefore, the follower position is determined by

$$X_n^m = \frac{1}{2}\left(X_n^m + X_n^{m-1}\right) + 0.01 \times LF \times \ crn_4 \tag{11}$$

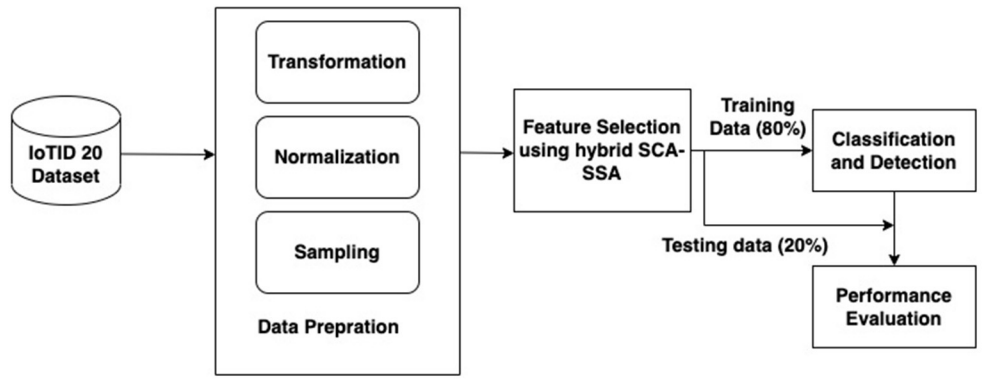


FIGURE 1
Proposed system framework.

TABLE 1 IoTID20 dataset instances.

| Category | Instances | Category | Instances | Category | Instances |
|---|---|---|---|---|---|
| Mirai | 415,677 | DoS | 59,391 | MITM ARP Spoofing | 35,377 |
| Scan | 75,265 | Normal | 40,073 | | |

TABLE 2 Metric formula.

| Metric | Formula | Metric | Formula |
|---|---|---|---|
| Accuracy | $\frac{(TPS+TNS)}{(TPS+FNS+TNS+FPS)}$ | Recall (R) | $\frac{TPS}{(TPS+FNS)}$ |
| Precision (P) | $\frac{TPS}{(TPS+FPS)}$ | F-Score | $\frac{2\times R\times P}{(R+P)}$ |

Where, FNS, false negative; TPS, true positive; TNS, true negative; FPS, false positive.

where, $crn_4$ is random number between [0,1], $X_n^m$ represents the $m$-th follower in $n$-th dimension. LF function is defined by following equation:

$$LF = \frac{f \times \sigma}{|g|^{\frac{1}{\beta}}}, \text{ where } \sigma = \left( \frac{\Gamma(1+\beta) \times \sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\left(\frac{\beta-1}{2}\right)}} \right)^{\frac{1}{\beta}} \quad (12)$$

where, β = 1.25. f and g are the random numbers, between [0, 1].

## 2.2.4. Classification and detection

With the feature subset obtained from the feature selection phase, classification and detection is performed using machine learning classifier. In this paper, multiclass classification is done i.e., detecting each attack category. In this paper, KNN and XGBoost classifier is used to identify intrusive network flow traffic. This sub-section describes these machine learning classifiers briefly.

### 2.2.4.1. KNN

The KNN is one the simplest supervised machine learning classifier. This classification method uses statistical measures to determine the instances proximity. K represents the number of neighbors. The data instances with high similarity belongs to same

1. Population initialization as i = 1,2,…,n
2. For each salp compute fitness function $X_n^{ite}$ = the best search agent
3. While($ite \leq ite_{\max}$)
4. Update LF, $crn_1$, $rnd_2$, $crn_4$, $rnd_3$, and $crn_3$
5. If ($ite$ ==1)
6. If ($crn_3 \geq 0.5$)
7. Use Eq. (9) to update salp position #SALP LEADER
8. else if ($crn_3 < 0.5$)
9. Use eq. (10) to update salp position #SALP LEADER
10. else if ($ite \geq 2$)
11. Use Eq. (11) to update salp positon #SALP FOLLOWER
12. end if
13. $ite = ite + 1$
14. end while
15. return optimal feature subset

Algorithm 1. Hybrid SCA-SSA

class. KNN allows new instances to be labeled using the previously labeled instances. However, it is highly sensitive to noise.

#### 2.2.4.2. XGBoost

This machine learning classifier allows to create sequential decision trees. This classifier is widely used for classification and regression task. XGBoost is used to speed-up the classification task and improving the classification performance. However, this classifier doesn't work well with unstructured data.

## 3. Result

The experiment is conducted on MAC Catalina OS with 8GB RAM. The implementation of classifiers and feature selection algorithms is done using python programming language. The formula of metrics used to evaluate performance is given in Table 2.

**Time** is the total computation time an IDS model will take with feature selection algorithm (in secs).

Number of features (N.F.) defines length of features selected by feature selection algorithm.

$$\text{Fitness Function} = \propto \times \left(1 - Accuracy\right) + \beta \times \frac{N.F.}{Maximum\ features} \text{ where, } \alpha = 0.99,\ \beta = 1 - \alpha$$

### 3.1. Result analysis

The proposed system is compared with the following algorithm such as sine-cosine algorithm (SCA) (Mirjalili, 2016), salp swarm algorithm (SSA) (Mirjalili et al., 2017), Jaya Algorithm (JA) (Rao, 2016), Bat algorithm (BA) (Yang, 2010), and grey wolf optimization algorithm (GWO) (Mirjalili et al., 2014), used for the feature selection task of IDS. Afterward the performance of two classifiers KNN and XGBoost is compared with these feature selection methods. Figure 2 depicts that the SCA-SSA with KNN and XGBoost attains the highest accuracy of 83.3% with the KNN classifier and 84.75% with the

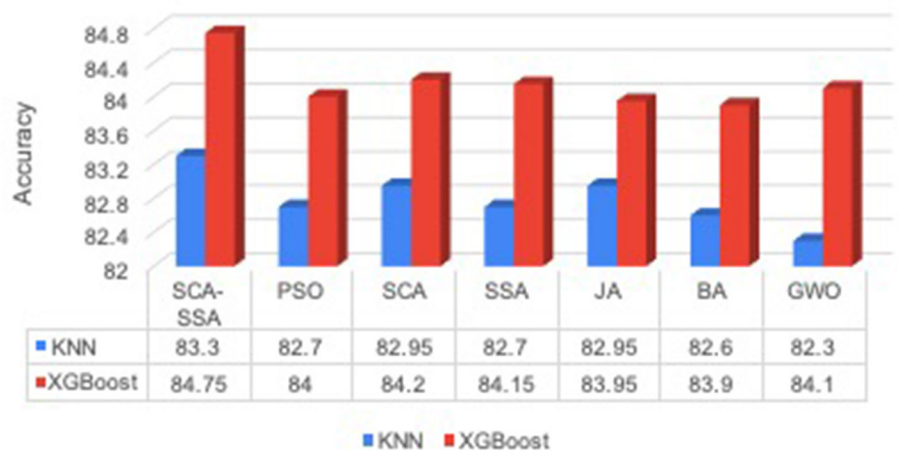


FIGURE 2
Accuracy.

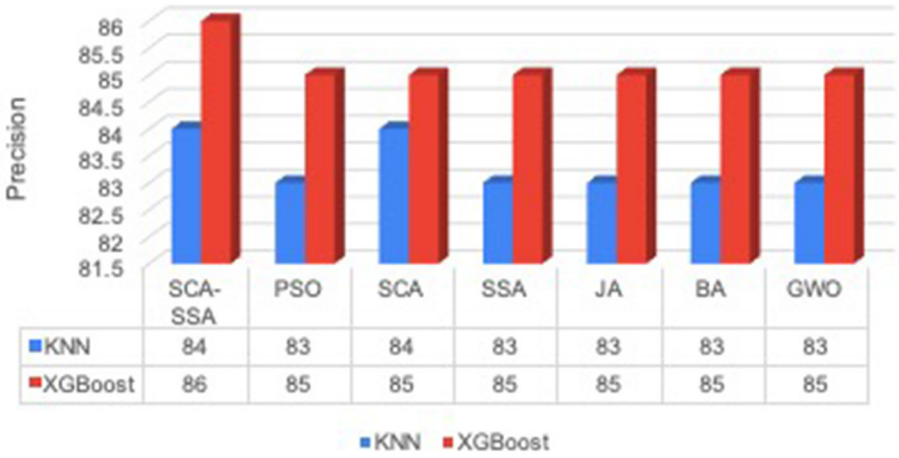


FIGURE 3
Precision.

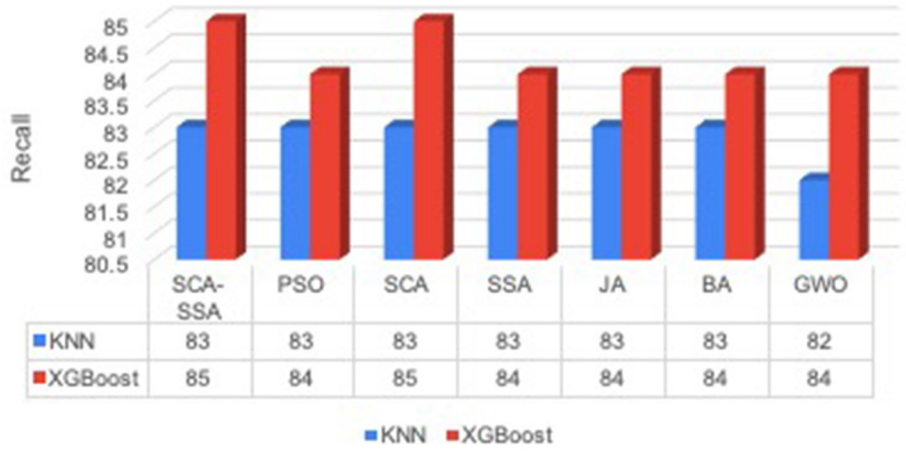


FIGURE 4
Recall.


FIGURE 5
*F*-score.


FIGURE 6
Time.


FIGURE 7
Feature selected.

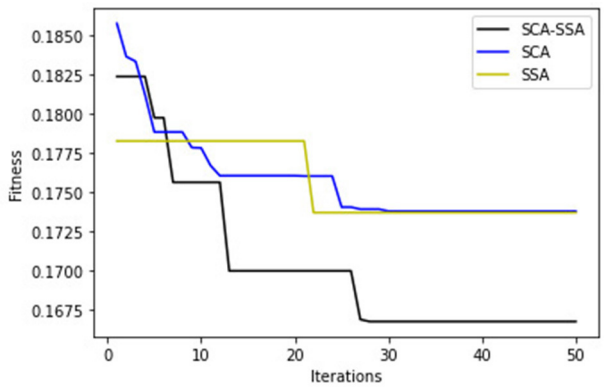


FIGURE 8
KNN-convergence curve.

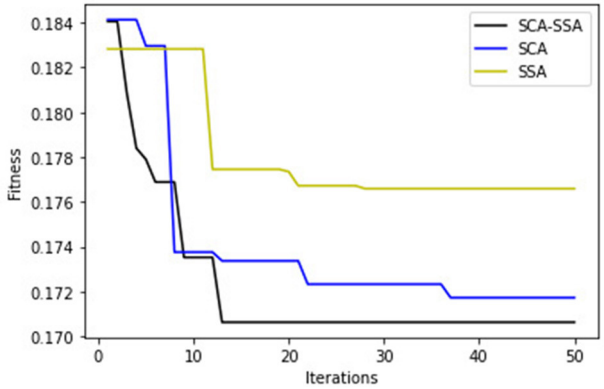


FIGURE 9
XGBoost-convergence curve.

XGBoost classifier. On the other side, BA attains the minimum accuracy of 82.6% with KNN and 83.9% with the XGBoost classifier.

Figure 3 depicts that SCA-SSA attains a precision of 84% with KNN and 86% with XGBoost, which is the highest among all other methods. Similarly, recall and F-Score of SCA-SSA are highest among other methods with KNN and XGBoost classifier as shown in Figures 4, 5, respectively. SCA-SSA takes the lowest execution time of 145.78 s with KNN, and 96.42 s with XGBoost as depicted in Figure 6. SSA takes the highest execution time with KNN, and GWO takes the highest execution time with the XGBoost classifier. Figure 7 shows SCA-SSA selects features with a minimal length of 11 with KNN and 8 with XGBoost classifier. Figures 8, 9 show the convergence curve of hybridized SCA-SSA with KNN, and XGBoost, respectively, which is better compared to convergence curve of original SCA, and original SSA. The overall performance of SCA-SSA with XGBoost classifier is better compared to SCA-SSA with KNN classifier in terms of all metrics used for performance evaluation of this system.

## 4. Discussion

Prevention of IoT devices from malicious activities is crucial. Data produced by these smart devices are heterogeneous. Hence, an IDS with an efficient feature selection method is important, with an objective of reducing the data size and detection time without compromising the system accuracy. The hybridized bio-inspired IDS for the IoT system is proposed in this paper. The IoTID20 dataset determines the performance of the proposed work. The optimal features are selected using hybridized SCA-SSA algorithm from pre-processed data. Then KNN and XGBoost classifiers are used to perform multiclass classification of intrusive data. Experimental result shows that the proposed system ameliorates other similar approaches. The convergence rate of SCA-SSA is better compared to SCA, and SSA. The hybrid SCA-SSA is compared against other metaheuristic algorithm used for feature selection tasks such as SCA, SSA, JA, BA, and GWO. The result shows that hybrid SCA-SSA with KNN and XGBoost attain high accuracy with the least execution time. Furthermore, the number of features selected is lowest with KNN and XGBoost

for hybrid SCA-SCA. Overall, the performance of SCA-SSA-XGBoost is better compared to SCA-SSA-KNN. In the future, we can use deep learning models for better performance of the proposed work.

## Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author/s. The IoTID20 dataset is analyzed for this research. This dataset is publically accessible at: https://sites.google.com/view/iot-network-intrusion-dataset/home.

## Author contributions

RS done paper drafting, implementation, and paper writing under the supervision of RU. All authors contributed equally in paper review. All authors contributed to the article and approved the submitted version.

## Funding

Guru Gobind Singh Indraprasth University is granting short term research fellowship to RS with reference to GGSIPU/DRC/2022/1218.

## Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

## Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

## References

Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., and Ayyash, M. (2015). Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Commun. Surv. Tutorials.* 17, 2347–2376. doi: 10.1109/COMST.2015.2444095

Alweshah, M., Hammouri, A., Alkhalaileh, S., and Alzubi, O. (2022). Intrusion detection for the internet of things (IoT) based on the emperor penguin colony optimization algorithm. *J. Ambient Intell. Hum. Comput.* 2022, 1–18. doi: 10.1007/s12652-022-04407-6

Balasaraswathi, V. R., Sugumaran, M., and Hamid, Y. (2017). Feature selection techniques for intrusion detection using non-bio-inspired and bio-inspired optimization algorithms. *J. Commun. Inf. Networks*. 2, 107–119. doi: 10.1007/s41650-017-0033-7

Chaabouni, N., Mosbah, M., Zemmari, A., Sauvignac, C., and Faruki, P. (2019). Network Intrusion Detection for IoT Security Based on Learning Techniques. *IEEE Commun. Surv. Tutorials*. 21, 2671–2701. doi: 10.1109/COMST.2019.2896380

Chawla, M., and Duhan, M. (2018). Levy flights in metaheuristics optimization algorithms—A review. *Appl. Artif. Intell.* 32, 802–821. doi: 10.1080/08839514.2018.1508807

Dahou, A., Elazis, M. A., Chelloug, S. A., Awadallah, M. A., Al-Betar, M. A., Al-qaness, M. A. A. et al. (2022). Intrusion detection system for iot based on deep learning and modified reptile search algorithm. *Comput. Intell. Neurosci*. 2022, 6473507. doi: 10.1155/2022/6473507

Davahli, A., Shamsi, M., and Abaei, G. (2020). Hybridizing genetic algorithm and grey wolf optimizer to advance an intelligent and lightweight intrusion detection system for IoT wireless networks. *J. Ambient Intell. Hum. Comput*. 11, 5581–5609. doi: 10.1007/s12652-020-01919-x

Fatani, A., Dahou, A., Al-qaness, M. A., Lu, S., and Elaziz, M. A. (2021). Advanced feature extraction and selection approach using deep learning and aquila optimizer for iot intrusion detection system. *Sensors*. 22, 140. doi: 10.3390/s22010140

Haddadpajouh, H., Mohtadi, A., Dehghantanaha, A., Karimipour, H., Lin, X., Choo, K. K. R. et al. (2020). A multikernel and metaheuristic feature selection approach for IoT malware threat hunting in the edge layer. *IEEE Internet Things J.* 8, 4540–4547. doi: 10.1109/JIOT.2020.3026660

Hosseini, F., Gharehchopogh, F. S., and Masdari, M. (2022). A botnet detection in IoT using a hybrid multi-objective optimization algorithm. *New Gener. Comput.* 40, 809–843. doi: 10.1007/s00354-022-00188-w

Hussain, F., Hussain, R., Hassan, S. A., and Hossain, E. (2020). Machine learning in iot security: current solutions and future challenges. *IEEE Commun. Surv. Tutorials*. 22, 1686–1721. doi: 10.1109/COMST.2020.2986444

Kareem, S. S., Mostafa, R. R., Hashim, F. A., and El-Bakry, H. M. (2022). An effective feature selection model using hybrid metaheuristic algorithms for IoT intrusion detection. *Sensors*. 22, 1396. doi: 10.3390/s22041396

Keserwani, P. K., Govil, M. C., Pilli, S. E., and Govil, P. (2021). A smart anomaly-based intrusion detection system for the Internet of Things (IoT) network using GWO–PSO–RF model. *J Reliable Intell. Environ*. 7, 3–21. doi: 10.1007/s40860-020-00126-x

Khraisat, A., and Alazab, A. (2021). A critical review of intrusion detection systems in the internet of things: techniques, deployment strategy, validation strategy, attacks, public datasets and challenges. *Cybersecurity*. 4, 1–27. doi: 10.1186/s42400-021-00077-7

Krishna, E. S., and Arunkumar, T. (2021). Hybrid particle swarm and gray wolf optimization algorithm for IoT intrusion detection system. *Int. J. Intell. Eng. Syst.* 14, 66–76. doi: 10.22266/ijies2021.0831.07

Mafarja, M., Heidari, A. A., Habib, M., Faris, H., Thaher, T., Aljarah, I. et al. (2020). Augmented whale feature selection for IoT attacks: Structure, analysis and applications. *Future Gener. Comput. Syst*. 112, 18–40. doi: 10.1016/j.future.2020.05.020

Mirjalili, S. (2016). SCA: A sine cosine algorithm for solving optimization problems. *Knowl. Based Syst.* 96, 120–133. doi: 10.1016/j.knosys.2015.12.022

Mirjalili, S., Gandomi, A. H., Mirjalili, S. Z., Saremi, S., Faris, H., Mirjalili, S. M. et al. (2017). Salp swarm algorithm: a bio-inspired optimizer for engineering design problems. *Adv. Eng. Software.* 114, 163–191. doi: 10.1016/j.advengsoft.2017.07.002

Mirjalili, S., Mirjalili, S. M., and Lewis, A. (2014). Grey wolf optimizer. *Adv. Eng. Softw.* 69, 46–61. doi: 10.1016/j.advengsoft.2013.12.007

Priya, R. M., Maddikunta, S., Parimala, P. K. R. M., Koppu, S., Gadekallu, T. R., Chowdhary, C. L. et al. (2020). An effective feature engineering for DNN using hybrid PCA-GWO for intrusion detection in IoMT architecture. *Comput. Commun.* 160, 139–149. doi: 10.1016/j.comcom.2020.05.048

Rao, R. V. (2016). Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems. *Int. J. Indus. Eng. Comput.* 7, 19–34. doi: 10.5267/j.ijiec.2015.8.004

Saif, S., Das, P., Biswas, S., Khari, M., and Shanmuganathan, V. (2022). HIIDS: Hybrid intelligent intrusion detection system empowered with machine learning and metaheuristic algorithms for application in IoT based healthcare. *Microprocess. Microsys.* 2022, 104622. doi: 10.1016/j.micpro.2022.104622

Sarwar, A., Alnajim, A. M., Marwat, S. N. K., Ahmed, S., Alyahya, S. S., and Khan, W. U. (2022). Enhanced anomaly detection system for IoT based on improved dynamic SBPSO. *Sensors*. 22, 4926. doi: 10.3390/s22134926

Thakkar, A., and Lohiya, R. (2021). A review on machine learning and deep learning perspectives of ids for iot: recent updates, security issues, and challenges. *Arch. Comput. Methods Eng.* 28, 3211–3243. doi: 10.1007/s11831-020-09496-0

Ullah, I., and Mahmoud, Q. H. (2020). "A scheme for generating a dataset for anomalous activity detection in IoT networks", in *Canadian Conference on Artificial Intelligence* (Cham: Springer), 508–520.

Yang, X. S. (2010). "A New Metaheuristic Bat-Inspired Algorithm", in *Nature inspired cooperative strategies for optimization (NICSO 2010)* (Berlin, Heidelberg: Springer), 65–74.