Received: May 23, 2021. Revised: July 12, 2021. 30

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Enhancing the Performance of an Intrusion Detection System Using Spider Monkey Optimization in IoT

Ethala Sandhya[1]* **Annapurani Kumarappan[1]**

[1]*Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur Campus, India*
* Corresponding author's Email: sandhyaethala@gmail.com

Abstract: The Internet of Things (IoT) has led to an era with the development of communication between smart devices used in various fields. Due to insufficient measures in the infrastructure of smart devices, it is prone to various attacks, which are launched by intruders during data transmission through the internet. Therefore, providing security measures for IoT systems is considered the highest priority. The present work ensures an Intrusion Detection System (IDS), which monitors malicious activities and helps in the successful functioning of IoT networks. Therefore, Spider Monkey Optimization with Random Forest (SMO-RF) algorithm is proposed to detect attacks in IoT environments. The NSL-KDD dataset is used where the input data is pre-processed and then classification is done using Random Forest (RF). The SMO will consider the features from where the decision for splitting or combining of the data taken by the female leader based on the 80-20 rule at the Global Leader Phase (GLP). Calculating the feature or variable importance with a Random Forest model shows which and all the features of the data are the most helpful for Classification is used. Thus, proposed SMO selects the best highest feature importance value for the classification of attacks using RF that overcomes the problem of over fitting. The results obtained from the proposed SMO-RF show the accuracy of 99.98 % better when compared with the existing SVM-parameter optimization and PSO-multi SVM techniques of 99.8% and 98 % respectively.

Keywords: Attacks, Internet of things, Intrusion detection system, NSL-KDD, Security, Spider monkey optimization with random forest.

1. Introduction

Internet of Things (IoT) is a developing technology that provides a unique connection with various devices to provide automatic operations and services in different fields ranging from daily life to critical infrastructure systems [1]. The goal of IoT is to connect the devices with the internet to create smart devices for storing, sharing, and gathering data without the interaction of humans [2]. As the numbers of IoT devices are deployed dramatically will also increase the volume of traffic in IoT devices, which reduces the unprecedented levels during DoS attacks. The timely detection of IoT Botnet has become an imperative strategy for risk mitigations that were associated with the attacks [3]. The IoT attacks are malicious and are needed to prevent them when accessing the legitimate users of the network and the system application software or information [4]. The integrity of models primarily aims for IoT attacks detection based on the violation occurred for the information in an IoT environment [5]. The occurrence of natural cause's errors such as systems poor configuration, services, and IoT attacks result in a violation for the e-services, networks, and resources [6]. The process of collecting malware and analyzing them is difficult as each iteration takes the malware's new characteristics without including the main codes and makes it hard to identify the simple patterns of matching. The malware's examples are available online for performing educational purposes, which are obtained by forensic examinations of infected machines [7]. The IoT attacks aim to corrupt the availability of particular nodes or the whole network

by jamming the signal or exhausting the resource nodes' battery. There are two kinds of IoT attacks such as, one that crashes the service and the other, which floods the service [8]. Initially, the IoT is characterized by attackers by an explicit attempt to prevent the services of legitimate use and deploys the various devices by preventing to attain the goals [9]. The IoT attacks are succeeded and it will consider the network server for utilizing the IoT devices which manages the results obtained from the network up to maximum industrial losses [10]. To solve such an issue, the proposed SMO-RF detects the IoT attacks in the devices using an optimization algorithm that overcomes the error loss factor during the attack detection. The complex optimization problems are applied and solved for obtaining a better solution. The main objective is to perform optimization to achieve the best set of prioritized constraints. The SMO will consider the features from where the decision for combining or splitting is taken care of by the female leader for selecting the best features for attack classification to find the best optimal solution. The selected 14 features are 'same_srv_rate','flag_SF','flag_S0','srv_serror_rate','serror_rate','dst_host_srv_serror_rate','dst_host_serror_rate','logged_in','dst_host_same_srv_rate','dst_host_same_src_port_rate','dst_host_srv_count','count','Protocol_type_icmp', 'service_eco_i'. The proposed research work considers features from the 16 features that were relevant for the feature selection process. The 14 features are selected using SMO at GLP based on 80-20 rule that identifies inputs that are potentially the most informative and make them the priority. The tree's decision is performed for each of the class objects that represent as a vote. The most important features during cross validation is decided by RF as it influences on the target variable. The forest selects the class, which has received a number of votes for the objects classifies the attacks into probe, R2L (Remote to Local), U2R (User to Root), and DoS (Denial of Service). The proposed SMO-RF showed an accuracy improvement of 0.12% to 0.5% of improvement compared to the existing models such as LSTM RNN with PSO and SMO-DNN techniques.

The organization of the paper is given as follows: Section 2 discusses the literature review of the existing methodologies and Section 3 discusses the proposed SMO-RF system. Section 4 describes the results and discussions and the conclusion of this research is presented in Section 5.

2. Literature review

The existing techniques based on intrusion detection are reviewed in this section.

Khraisat [11] developed hybrid IDS by combining the C_5 classifiers and the Support Vector Machine (SVM) classifier. The higher rate of detection reduced the false positive rate and the developed SVM model was not adequate in detecting the various attacks in IoT systems, which required higher memory, and processing consumption. Ravi [12] developed a mitigation approach for DDoS attack detection in IoT through the SDN Cloud architecture. The developed approach prevented the denial of services to users when the server of IoT was attacked by the wireless IoT. The developed learning-driven detection approach showed lesser performance in detecting the attacks as training of data was poor due to usage of a small dataset.

Ko [13] established features dynamic deep learning model for the mitigation of DDoS within the domain of Internet Service Providers (ISP). The features dynamic method enhanced the performance of feature selection. The developed features dynamic method was computationally more expensive. Monika Roopak [14] developed multiple objective-based feature selection approach for the detection of DDoS attack in IoT. The multiple objective methods showed higher accuracy in selecting the features. The developed algorithm selected only a limited number of features due to which performance in terms of accuracy was decreased.

Tuan [15] presented the botnet DDoS attack detection approach by utilizing machine learning methods like Support Vector Machine (SVM), Decision Tree (DT), Naïve Bayes (NB), Artificial Neural Network (ANN), K-means, and X-means. The developed method effectively classified the botnet and normal network traffics. The ANN has not accurately detected the DDoS attacks due to almost flattened curves of the traffic classes.

Kumar [16] developed IDS using the Multi Linear Dimensionality Reduction (ML-DR) with Multi-Class SVM. The combined dimensionality reduction technique with the Multi-class SVM reduced the dimension as well as shortens the training time. However, the rank limitation during dimensionality reduction was shown in discriminant vectors that hindered the information obtained classification results poor.

Setiawan [17] developed IDS by fusing the normalization, feature selection, Support Vector Machine (SVM) for the classification. The Modified Rank-based information gain feature selection model, and normalization process with SVM trained the unbalancing data. The developed model combined Modified rank based information gain feature selection model and improved the accuracy of the model but the model created under fitting problems

due to inefficient optimization process as 17 parameters alone were utilized for feature selection

Zhou [18] developed an efficient IDS using Correlation-based feature selection and Bat Algorithm (CFS-BA) with ensemble classifier for attack classification. The ensemble approach combined the RF that Penalized Attributes (PA) as a voting technique combined the probability distributions for attack recognition. However, intrusion detection was not claimed but showed better performance compared to another context for intrusion detection.

Neelu Khare [19] developed SMO -Deep Neural Network Hybrid Classifier Model for IDS. The developed SMO algorithm was used for dimensionality reduction and the reduced dataset was fed into a deep neural network (DNN). The results justified the advantage of implementing the developed model over other approaches reduced the dimensionality problem. The limitation of the developed model was that it was only applied for binary classification.

Elmasry [20] developed deep learning architectures for network intrusion detection using a double PSO meta-heuristic. The developed model utilized double Particle Swarm Optimization (PSO)-based algorithm for selecting both feature subset and hyper parameters in one process from where the optimal feature subset was selected. The hyper parameters were optimized and maximized the accuracy. However, only few types of attacks were included in the test set rather than in the training set to examine the ability that failed to classify them properly.

3. Proposed methodology

The flow diagram of the proposed SMO-RF method is shown in Fig. 1. The data is collected from the NSL-KDD dataset and undergoes pre-processing to remove the unwanted data. The proposed SMO-RF, SMO will consider the features from where the female leader decides whether to split or combine thereby selects the best highest feature importance value for classification of attacks using RF.

3.1 Data collection

The NSL-KDD dataset is used in the present research work. The NSL KDD dataset is the common dataset used in the environment of IoT. It is formed from the various part of KDDcup 99 without any duplications and redundancy. The NSL KDD dataset contains 41 attributes that are labeled attack types and normal connections. The dataset solves the inherent problems as it consists of 4 types of attacks i.e. probe, R2L (Remote to Local), U2R (User to Root), and DoS (Denial of Service).

Probe attack: The probe attack is affected due to misused information that weakens the strength of the network. The probe attacks include Portsweep, Satan, Ipsweep, Mscan, Saint, and Nmap.

R2L: By transmitting the packets to the machine from the user detects the weakness of the network. There are several attacks such as Send mail, Snmpget attack, Snmpguess, Phf, Warez client, Ftp-write, Guess-Password, Multihop, Xsnoop, Httptunnel, Xlock, Spy, Warezmaster, and Imap.

U2R: The U2R gets access to the root account once the setup is done for an ordinary account. These attacks in U2R include Perl, Buffer-overflow, Load module, Rootkit, Sqlattack, Xterm, and Ps.

DoS: The DoS attacks are increased due to the traffic usage in the network that is not provided by the system for DoS attacks detection. The types of attacks present in DoS include Apache2, Neptune, Udp storm, Land, Back, Teardrop, Smurf, Pod, and worm. Table 1 shows Statistical information about the NSL-KDD dataset [21].

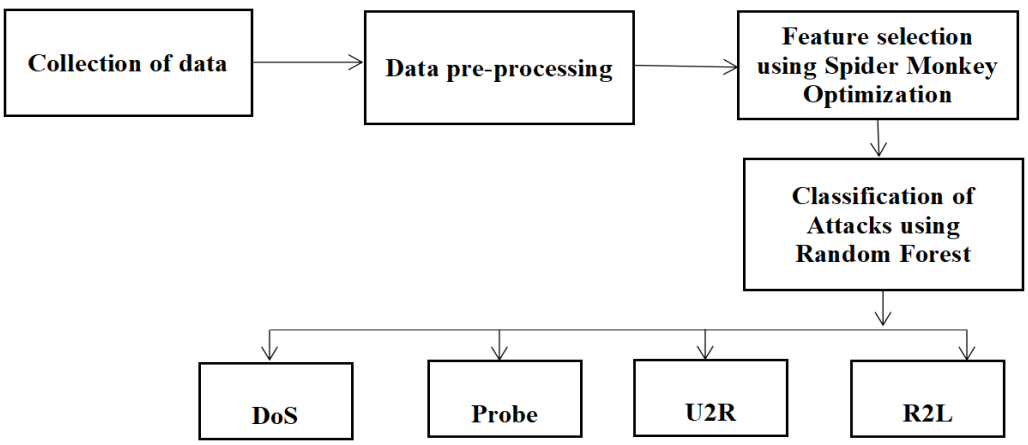

Figure. 1 The flow diagram of the proposed SMO-RF for attack detection

Table 1. Statistical information about the NSL-KDD dataset

Abnormal					**Normal**	**Total**
Dataset	DoS	R2L	Probe	U2R		
KDD Train +	45927	995	11656	52	67343	125973
KDD Test+	7458	2421	2754	200	9711	22544

The original network traffic records in the NSL-KDD dataset are stored as 41-dimensional vectors, containing both numerical values and categorical values and these values are fed to the pre-processing stage to normalize the data.

3.2 Data pre-processing

After collecting the datasets, the input data is pre-processed to remove the missing data and noises. The pre-processing techniques such as data cleaning, normalization, integration, and transformation of data were utilized.

The data cleaning process is performed for the data that creates the data for modifying or deleting the incorrect data, duplicate data, irrelevant data and improper data. The data cleaning is about not only information erasing but also making space for the new data and find a way for data set accuracy maximization without information deletion. The data cleaning includes more removal of data such as spelling and syntax errors and datasets standardization and correcting those mistakes for clearing the missing codes, empty fields, identifying the data points that were duplicates. The main goal is to clean the data and create a dataset of the standardized ones and overcoming those problems is important.

Next, the process of the normalization process using the Min-Max technique is performed that integrates and normalizes data. The data were converted to a particular format by aggregating, smoothing and lower level of raw data are replaced by the higher-level data by using the concept of hierarchy. The integration of data combines the residing data in various sources and provides a unified view to users. The examination of a large data is considered for the present research to determine the attacks accurately. The missing data in a large dataset occurs due to human error, database failure, or malfunctioning of the system. To avoid this, the missing data will be filled by the structured data. As the incomplete or uncertain data present these missing data should be modified by deleting unwanted data to improve its quality. Next, the integration of data is done by data pre-processing. The Min-Max normalization process plays an important role in the integration and as well as data normalization. Each feature value that is having a minimum value gets transformed into 0 and the maximum value is transformed into 1. All the values will be converted from decimals ranging from 0 and 1 and these data are integrated to form the pre-processed data. The pre-processed data are used for feature selection using the SMO algorithm to overcome the problem of optimization.

3.3 Feature selection process using spider monkey optimization algorithm

SMO is a meta-heuristic technique designed based on spider monkeys foraging behaviour. Each spider's foraging behaviour is determined based on social structures such as fission and fusion processes. The features of the algorithm are dependent on a group where the decision is taken care of by the female leader, one who decides to split or combine the group [22].

- The whole group of spider monkeys has a leader who is known as the global leader and under it, small groups of spider monkeys have leaders in each group who are known as local leaders.
- The SMO is a swarm intelligence technique and each small group consists of a minimum number of monkeys. If any further fission is created, then it creates one group with a minimum number of monkeys and it is defined as the time for fusion.
- The SM in the SMO is having a potential solution that can be represented with the Local Leader phase, second as Global leader phase, third as Local leader learning phase, fourth as Global Leader Learning phase, fifth as Local Leader Decision phase and six as Global Leader Decision phase.

3.3.1. Initializing the population

The SMO is distributed with the population $P = 50$ and SM_p $(wherep = 1, 2 \ldots P$)and SM_p is known as the population for the p^{th} monkey. Consider the monkeys are present as M-dimensional vectors and M is known as the total number of variables for overcoming the problem. Each SM_p obtains a possible solution for the obtained problem and each of $SM_p$is initialized as shown in Eq. (1):

$$SM_{pq} = SM_{minq} + UR(0,1) \times (SM_{maxq} - SM_{minq}) \quad (1)$$

where, SM_{pq} is known as the p^{th} SM having q^{th} dimension.

SM_{minq} and SM_{maxq} are known as the lower bounds and upper bounds moving in q^{th} direction and the value of $q = 1,2 \ldots M$. The random number is represented as $UR(0,1)$ which is uniformly distributed that ranges from [0, 1].

3.3.2. Local leader phase (LLP)

The present location of SM is changed and utilizes the past occurrences in LLP for both the local group members and local leaders. The new location is updated as a new SM location is assigned in such a way that the fitness values get higher compared to the previous locations. The location is updated for the p^{th}SM having l^{th} a local group is given by using Eq. (2).

$$SM_{newpq} = SM_{pq} + UR(0,1) \times \left(LL_{lq} - SM_{pq}\right) + UR(-1,1) \times (SM_{rq} - SM_{pq}) \quad (2)$$

Where LL_{lq} is having q^{th} dimension from the randomly chosen l^{th}SM of a l^{th}alocal group having their leader location. The SM_{rq} is having q^{th} a dimension that randomly chooses l^{th} SM of l^{th} local group where $r \neq p$.

3.3.3. Global leader phase (GLP)

The GLP is initialized once after the LLP is processed. The location of the SM is calculated using Eq. (3) and the location updated is calculated as shown as follows.

$$SM_{newpq} = SM_{pq} + UR(0,1) \times \left(GL_{lq} - SM_{pq}\right) + UR(-1,1) \times (SM_{rq} - SM_{pq}) \quad (3)$$

Where GLq is defined as the global leader location having q^{th} dimension ($q = 1, 2, 3, \ldots M$) having the arbitrarily selected index. The SM fitness is calculated using the probability $prbp$. Based upon the probable value, the location of SM_p location is updated. The better location candidates are having access for number of possibilities increased makes better. Therefore, the proposed research work considers 14 features from the 16 features that were relevant out of a total of 104 features that are used for the feature selection process. The 14 features are selected using SMO based on 80-20 rule that identifies inputs that are potentially the most informative and make them the priority gives access for number of possibilities. The Rule based approach in SMO works at GLP phase that selects the informative features are accessed upon the number of possibilities. The Rule based SMO makes a priority which is based upon the condition that if $\sum_{i \epsilon S} fitness_{feature_{importance}} \geq 0.8$, where S is the selected feature value obtained is utilized for feeding the data to the tree, else the features are not considered for classification of attacks. The fitness function having the highest feature value is obtained based upon the condition considered. The calculation of the probability is followed using Eq. (4).

$$prbp = f_{n_p} \sum N_p = 1 f_{n_p} \quad (4)$$

Where f_{n_p} is known as the p^{th} SM fitness value and further fitness value for the new location are calculated using Eq. (4) and compares the value with the old location. The location is going to be adopted based on the fitness value function.

3.3.4. Global leader learning (GLL) phase

The GLL stage will be undergone and updated based on the greedy selection method. The best fitness value is generated and selected from the population. An optimum location is obtained that value will be assigned for the global leader. If there is no update performed, increment a value of 1 to the GlobalLimitCount.

3.3.5. Local leader learning (LLL) phase

The greedy selection model is applied for a local group to update the LLL with the SM location based on the fitness value for that particular local group. An optimum location is having a value that is assigned to the local leader. If there is no further updation present, increment it to 1 that will be added for the LLC.

3.3.6. Local leader decision (LLD) phase

If the local leader is not updated for their location among the fixed LLL then the local group present are having the candidates modified with the random location from step 1 that utilizes the information from the global leader and local leader based on prusing the Eq. (5).

$$SM_{newpq} = SM_{pq} + UR(0,1) \times (GL_{lq} - SM_{pq}) + UR(0,1) \times \left(SM_{rq} - LL_{pq}\right) \quad (5)$$

3.3.7. Global leader decision (GLD) phase

The global leader will be not updated with its location based on the GLL and then the population will be split up into small-sized groups according to the GLD. The group is split up into the process and

continues till the maximum number of groups (MG) is allowed. Once the groups are created with a global leader, the GLD will fail for position updation until the prefixed limits were allowed. Then the global leader will be merged with an entire group forms a single group. The total number of iterations are considered to be as 10 and the local leader selects the newly created shaped groups. The maximum number of allowed groups were created and the GL fails for position updation until the prefixed is allowed with limits. It decides whether to merge entire single groups or to merge into a single group. As the SMO is a population-based algorithm it is inspired mainly based on the social actions of the SMs provides higher efficiency. They finally deal with the variants successfully for real world optimization problems using fitness function. The fitness value is calculated using the feature importance that assigns a score for each input features based on the target variable prediction. The feature importance is calculated as the node impurity weights are decreased before the probability of reaching the node. The node probability calculates ratio of the number of samples that reaches the nodes to that of the total number of samples. The best value of the feature is selected based on the highest value of the fitness function using Eq. (6).

$$fitness_{feature_{importance}} = \frac{\text{Number of samples that reaches the nodes}}{\text{Total number of samples}} \quad (6)$$

The best values obtained are fed to the RF classifier to perform the classification of attacks. Therefore, the SMO will reduce the overfitting problems and the informative features ϵS is fed to RF for classifying the attacks.

3.4 Classification using random forest

Once the highest fitness function is selected as an optimum value, the obtained optimal values are fed into the RF classifier for the classification of attacks. The Random forest classifier is having decision trees and therefore lower classification error is present when compared with the existing classification algorithms. An advantage of the Random Forest classifier in the research work is that the important features automatically classify the attacks as Dos, Probe, R2L, U2R.

The tree's decision is performed for each of the class objects that represent as a vote. The forest selects the class, which has received a number of votes for the objects. Therefore, RF utilizes both boosting and bagging as the successful approach select the random variable for building the tree. The features present in the random forest are explained as follows:

Using the Random forest, the generalization error is bound to be dependent mainly on the tree strength that achieves correlation among them. Based on the maximum voting approach, the elements such as i and j are voted in the RF model thereby classifies the attacks using the following Eq. (7).

$$prox(i,j) = \frac{\sum_{t=1}^{ntree} 1(h_t(i)=h_t(j))}{ntree} \quad (7)$$

where $1(\cdot)$ represents the indicator function,

h_t represents the tree of the forest

$h_t(i)$is the value which is predicted for all the values of i

If $prox(i,j)$ = 1 then the classes $i$and j of the same classes are classified

4. Results and discussion

The proposed SMO-RF method is simulated using Anaconda navigator and python 3.6 software with the system requirements; operating system: windows 10, RAM: 128 GB, Processor: Intel Core i9 with 3GHz, and hard disk: 4 TB. The present research work uses the NSL-KDD dataset for performing the testing.

The proposed SMO-RF method performance is evaluated by using the following parameters.

Accuracy:

Accuracy is the measure used to predict the exactness of the machine learning model. The equation for accuracy is shown in Eq. (8).

$$Accuracy\% = \frac{TP+TN}{TP+TN+FP+FN} \times 100 \quad (8)$$

Precision:

Precision is the ratio of correctly predicted positive observation to the total predicted positive observation. The equation for Precision is shown in Eq. (9).

$$Precisison\ \% = \frac{TP}{TP+FP} \times 100 \quad (9)$$

Recall:

The Recall quantifies the number of positive class predictions made out of all positive examples in the dataset which is shown in Eq. (10).

$$Recall\% = \frac{TP}{TP+FN} \times 100 \quad (10)$$

F-Measure:

F-Measure provides a single score that balances both the concerns of precision and recall in one number. The equation for F-Measure is shown in Eq. (11).

$$F - Measure\ \% = \frac{2PR}{P+R} \times 100 \qquad (11)$$

Area Under Curve (AUC):

Area Under Curve (AUC) is defined as an area obtained by integrating between the two points. The area under the curve is represented as $y = f(x)$ between $x = a$ and $x = b$, limits of a and b.

Where,

P= Precision

R= Recall

TP=True Positive

TN=True Negative

FP= False Positive

FN= False Negative

4.1 Quantitative analysis

Table 2 shows the results obtained by using the proposed SMO-RF in terms of Accuracy, Precision, Recall, F-measure and AUC. The SVM uses the hyperplane parameters for tuning and fails to perform well when the data set has more target classes overlapped for the NSL-KDD dataset. Whereas, the RF algorithm use data that considers a set of features irrespective of various scales that shows better performances. The RF is intrinsically suited for multiclass problems, that depend on the dataset and also considers many other systematic factors. Table 3 shows the results obtained for the proposed SMO-RF and without using SMO for performing classification. The results showed an average improvement of 0.5 % of performance which was seen among all the classes belonged to SVM, RF, and SMO-RF. The proposed SMO-RF utilizes the selected features that were fed and showed reasonable improvement. The graphical representation of the results obtained for the proposed method is shown in Fig. 2. The importance of feature selection for classification is shown in Fig. 3.

The proposed SMO-RF shows better results when compared to the other classifiers such as SVM and RF for the NSL-KDD dataset is as shown in Table 2. The graphical representation of the results obtained for the proposed SMO-RF compared with SVM and RF classifiers in terms of accuracy, precision, recall, F-measure and AUC is obtained as shown in Fig. 3.

Table 2. Results obtained for the proposed SMO-RF

Classifiers	Attacks	Accuracy (%)	Precision (%)	Recall (%)	F-measure (%)	AUC (%)
SVM	DoS	97.6	99.66	95.45	97.47	98.2
	Probe	98.84	90.54	90.64	91.43	92.56
	R2L	99.2	98.95	98.91	98.65	98.73
	U2R	99.62	99.87	99.86	99.81	98.4
RF	DoS	98.4	99.83	96.68	98.31	98.43
	Probe	99.1	92.35	93.52	93.91	94.65
	R2L	99.56	98.83	98.73	99.12	99.43
	U2R	99.75	99.91	99.86	99.89	98.2
Proposed SMO-RF Method	DoS	98.9	99.97	99.97	99.97	98.6
	Probe	99.9	99.9	99.9	99.9	99.9
	R2L	99.93	99.99	99.93	99.91	99.91
	U2R	99.97	99.96	99.97	99.95	98.8

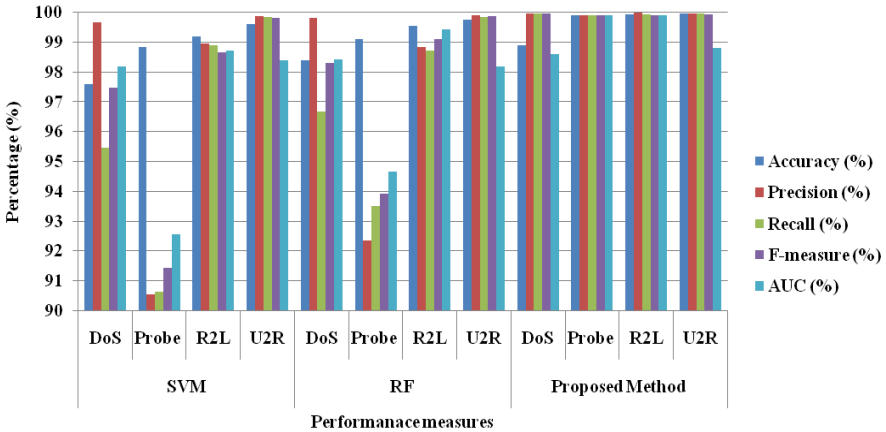

Figure. 2 The performance measures representation of the results obtained for the proposed SMO-RF

Table 3. The results obtained by using RF without SMO and RF with SMO

Classifier	Without SMO				With SMO			
Attacks	**DoS**	**Probe**	**R2L**	**U2R**	**DoS**	**Probe**	**R2L**	**U2R**
Accuracy (%)	98.4	99.1	99.56	99.75	98.9	99.9	99.93	99.97
Precision (%)	99.83	92.35	98.83	99.91	99.97	99.9	99.99	99.96
Recall (%)	96.68	93.52	98.73	99.86	99.97	99.9	99.93	99.97

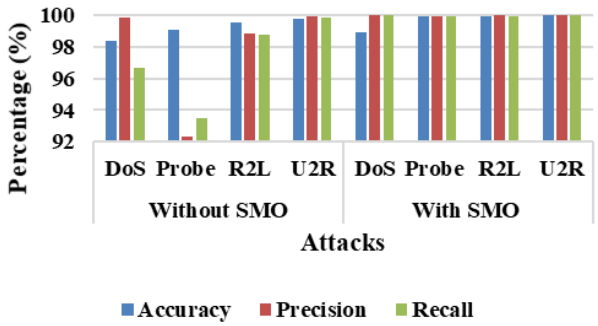

Figure. 3 The performance measures representation of the results obtained for with and without SMO for classification using RF

The performance measures obtained for the RF classifier with and without the SMO algorithm is shown in fig 3. Table 3 shows that the rate was increased by using SMO as it could extract the prominent features improved the classification of attacks.

4.2 Comparative analysis

The proposed SMO-RF has obtained better accuracy of 99.98 % of average accuracy for Multi-class, which showed better when compared with the existing SVM, ML-DR with SVM and CFS-BA-Ensemble method obtained an accuracy of 98.8%, 98 %, and 99.8 %, similarly SMO-DNN obtained 99.4 % and LSTM-RNN with PSO obtained 98.8 %. Table 4 shows the comparative analysis for the proposed SMO-RF with the existing methods.

Kumar [16] developed ML-DR with multi-class SVM showed classification results lowered due to non-consideration of discriminant vectored feature using PCA re. Therefore, the discriminant vectored features that showed optimization problem was overcome by the developed CFS-BA-Ensemble model [18]. However, the BA failed to make decisions due to the absence of local and global leader during the optimization process. These problems are overcome by the proposed SMO-RF model, where SMO constituted of global and local leaders for decision making in the selection of optimal features reduced the feature dimension as well as improved the performance in terms of accuracy when compared to the existing models. Similarly, Bambang Setiawan [17] showed lower accuracy values for the minority classes (R2L and U2R) as 17 features for data samples were utilized for training and testing during classification showed overfitting problems. Neelu Khare [19] showed only binary classification using SMO-DNN and failed to extend for detecting anomalies in IoT environments. Wisam Elmasry [20] classified only few types of attacks for the testing set rather than in the training set examined but the ability to classify the attacks were failed. Therefore, the proposed SMO-RF extensively trained the data that utilized the data for training and for data estimation. The SMO performs global optimization task well for constrained parameters that uses 14 informative features and reduced the dimensions of the classification algorithm overcame the overfitting problems. The SMO classified efficiently the data based on several intrusions and improved the system and classification performance using RF.

Table 4. Comparative analysis for the proposed SMO-RF with the existing methods

Authors	**Methodology**	**Accuracy (%)**
Bukka Narendra Kumar [16]	ML-DR with multi SVM	98
Bambang Setiawan[17]	SVM with parameter optimization	99.8
Yuyang Zhou [18]	CFS-BA-Ensemble method	99.8
Neelu Khare [19]	SMO-DNN	99.4
Wisam Elmasry [20]	LSTM-RNN with PSO	98.8
Proposed	SMO-RF	99.98

5. Conclusion

The DDoS attacks in IoT environments occur because of the lack of strong security monitoring and protection techniques. The proposed Spider Monkey Optimization with Random Forest (SMO-RF) algorithm detects the attacks in IoT devices. The purpose of optimization is to achieve the "best" design relative to a set of prioritized criteria or constraints. The SMO will consider the features from where the female leader from SMs decides whether to split or combine thereby selects the best features for the classification of attacks using RF. Based upon the 80-20 rule, the priority is given for each of the features and the selected feature value obtained is fed into the tree of RF. The fitness function which is having the highest value of feature will be considered as the best and the highest feature value is fed for the RF classifier. The overfitting problems were overcome as the SMO extracts the informative features alone that are used for classification. The proposed SMO-RF showed an accuracy improvement of 0.12% to 0.5% of improvement compared to the existing models such as ML-DR with multi SVM, SVM with parameter optimization, CFS-BA-Ensemble, LSTM-RNN with PSO and SMO-DNN techniques. In the future, the complexity of the system can be improved for better performance and results.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

The paper background work, conceptualization, methodology, dataset collection, implementation, result analysis and comparison, preparing and editing draft, visualization have been done by first author.

The supervision, review of work and project administration, has been done by second author.

References

[1] Y. Jia, F. Zhong, A. Alrawais, B. Gong, and X. Cheng, "Flowguard: an intelligent edge defense mechanism against IoT DDoS attacks", *IEEE Internet of Things Journal*, Vol. 7, No. 10, pp.9552-9562, 2020.

[2] C. D. McDermott, F. Majdani, and A. V. Petrovski, "Botnet detection in the internet of things using deep learning approaches", In: *Proc. of 2018 International Joint Conference on Neural Networks (IJCNN)*, pp. 1-8, 2018.

[3] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, A. Shabtai, D. Breitenbacher, and Y. Elovici, "N-baiot—network-based detection of IoT botnet attacks using deep autoencoders", *IEEE Pervasive Computing*, Vol. 17, No. 3, pp.12-22, 2018.

[4] Y. Imamverdiyev and F. Abdullayeva, "Deep learning method for denial of service attack detection based on restricted boltzmann machine", *Big Data*, Vol. 6, No. 2, pp.159-169, 2018.

[5] M. Habib, I. Aljarah, and H. Faris, "A Modified Multi-objective Particle Swarm Optimizer-Based Lévy Flight: An Approach Toward Intrusion Detection in Internet of Things", *Arabian Journal for Science and Engineering*, Vol. 45, No. 8, pp.6081-6108, 2020.

[6] S. D. Bhosale and S. S. Sonavane, "Design of Intrusion Detection System for Wormhole Attack Detection in Internet of Things", *Advanced Computing and Intelligent Engineering*, pp. 513-523, 2020.

[7] L. E. S. Jaramillo, "Malware Detection and Mitigation Techniques: Lessons Learned from Mirai DDOS Attack", *Journal of Information Systems Engineering & Management*, Vol. 3, No. 3, pp.19, 2018.

[8] A. Aldaej, "Enhancing cyber security in modern internet of things (IoT) using intrusion prevention algorithm for IoT (ipai)", *IEEE Access*, 2019.

[9] J. Li, M. Liu, Z. Xue, X. Fan, and X. He, "RTVD: A Real-Time Volumetric Detection Scheme for DDoS in the Internet of Things", *IEEE Access*, Vol. 8, pp. 36191-36201, 2020.

[10] C. A. D. Souza, C. B. Westphall, R. B. Machado, J. B. M. Sobral, and G. D. S. Vieira, "Hybrid approach to intrusion detection in fog-based IoT environments", *Computer Networks*, Vol. 180, pp.107417, 2020.

[11] A. Khraisat, I. Gondal, P. Vamplew, J. Kamruzzaman, and A. Alazab, "A novel ensemble of hybrid intrusion detection system for detecting internet of things attacks", *Electronics*, Vol. 8, No. 11, pp.1210, 2019.

[12] N. Ravi and S. M. Shalinie, "Learning-driven detection and mitigation of DDoS attack in IoT via SDN-cloud architecture", *IEEE Internet of Things Journal*, Vol. 7, No. 4, pp.3559-3570, 2020.

[13] I. Ko, D. Chambers, and E. Barrett, "Feature dynamic deep learning approach for DDoS mitigation within the ISP domain", *International Journal of Information Security*, Vol. 19, No. 1, pp.53-70, 2020.

[14] M. Roopak, G. Y. Tian, and J. Chambers, "Multi-objective-based feature selection for

DDoS attack detection in IoT networks", *IET Networks*, Vol. 9, No. 3, pp.120-127, 2020.
[15] T. A. Tuan, H. V. Long, R. Kumar, I. Priyadarshini, and N. T. K. Son, "Performance evaluation of Botnet DDoS attack detection using machine learning", *Evolutionary Intelligence*, pp.1-12, 2019.
[16] B. N. Kumar, R. MSVSB, and B. V. Vardhan, "Enhancing the performance of an intrusion detection system through multi-linear dimensionality reduction and Multi-class SVM", *International Journal of Intelligent Engineering and Systems*, Vol. 11, No. 1, pp.181-192, 2018.
[17] B. Setiawan, S. Djanali, T. Ahmad, and I. T. S. Nopember, "Increasing accuracy and completeness of intrusion detection model using fusion of normalization, feature selection method and support vector machine", *International Journal of Intelligent Engineering and Systems*, Vol. 12, No. 4, pp. 378-389, 2019.
[18] Y. Zhou, G. Cheng, S. Jiang, and M. Dai, "Building an efficient intrusion detection system based on feature selection and ensemble classifier", *Computer Networks*, Vol. 174, pp .107247, 2020.
[19] N. Khare, P. Devan, C. L. Chowdhary, S. Bhattacharya, G. Singh, S. Singh, and B. Yoon, "Smo-dnn: Spider monkey optimization and deep neural network hybrid classifier model for intrusion detection", *Electronics*, Vol. 9, No. 4, pp. 692, 2020.
[20] W. Elmasry, A. Akbulut, and A. H. Zaim, "Evolving deep learning architectures for network intrusion detection using a double PSO metaheuristic", *Computer Networks*, Vol. 168, pp. 107042, 2020.
[21] NSL-KDD data set, available on: http://nsl.cs.unb.ca/NSL-KDD/, 2013
[22] H. Sharma, G. Hazrati, and J. C. Bansal, "Spider monkey optimization algorithm", In: *Proc. of Evolutionary and Swarm Intelligence Algorithms*, pp. 43-59, 2019.