International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Optimized Maxout Classifier for Detection of DDoS Attack in SDN

Karthika P[1], Karmel A[*2]

Submitted: 10/09/2022 **Accepted:** 20/12/2022

Abstract: The SDN has increased its focus, and the notion of network control has offered an efficient network-oriented DDoS protection in addition to many DDoS assault methods. More information about the network might be influenced by the centralized SDN controller, and SDN framework helps in identifying DDoS assaults using various methods. The simulation dataset for this work was generated by constructing SDN on the Mininet emulator. To construct the dataset and train the deep learning algorithm, the unique features are logged into a csv file. Further, detection is done using Optimized Deep Max out classifier. In addition, the weights of Deep Max out classifier are chosen via Sine Map Insisted CA (SMI-CA) model. If any attack is found, Bait oriented mitigation is made for relieving from attacks. As last step, analysis is done to portray the effectiveness of adopted model. The model used in the paper is further evaluated using the newly released dataset CICDDoS2019 along with the simulation dataset. Result shows that the Deep Maxout classifier has a very low false alarm rate and can classify traffic with the greatest testing accuracy of 96.5% for the CICDoS2019 dataset and 95.1% for the simulation dataset.

Keywords: *Software-Defined Networking; DDoS Attack Detection; Coot Algorithm; Deep Maxout; SMI-CA Model*

Nomenclature

Abbreviation	Description
CIC	Canadian Institute of Cyber Security
CA	Coot Algorithm
CNN	Convolutional Neural Network
DCNN	Deep CNN
DBN	Deep Belief Network
DMO	Deep Max Out
DDoS	Distributed Denial Of Service
IDS	Intrusion Detection System
GA	Genetic Algorithm
KPCA	Kernel Principal Component Analysis
KNN	K-Nearest Neighbours
LEDEM	Learning Driven Detection Mitigation
LP	Learning Percentage
ML	Machine Learning
RNN	Recurrent Neural Network
SSA	Salp Swarm Algorithm
SVM	Support Vector Machine
SMO	Spider Monkey Optimization
SMI-CA	Sine Map Insisted CA
SDN	Software Defined Network
TOA	Teamwork Algorithm

1. Introduction

The services of networks with crucial industry and business data were spread to the manufacturing and life of contemporary society as a result of the ongoing development of communication expertise, the endless growth of internet production requirements, and the rapid expansion of the Online business in the Era of the internet [6] [7]. The beginning of DDoS assault could cause anomalies in the connected network services, ensuing in important financial loss and potentially other terrible effects. One of the major dangers to network security that the Internet is subject to is DDoS assaults. Accurate and rapid DDoS attack detection is a chief research area in security sector [8] [9]. "SDN is an emerging network innovation architecture that separates the network data plane and the control plane, which has the characteristics of network programmable, centralized management control, and interface opening [10] [11]".

In order to carry out DOS attacks, system attackers target varied resources [12] [13]. DDoS assaults demonstrate the rising size of the attack and the sophistication of the attack strategy. The following factors make it very hard to mark out the basis of an attack: (1) attack traffic characteristics that are hard to recognize; (2) need of cooperation among rational network node; (3) strengthening of attacking tool with a decreasing threshold of usage; (4) widely used address fraud; (5) short attack duration and limited response time [14] [15].

The two primary DDoS attack detection technologies in the conventional network architecture are attack identification depending on the traffic features and detection systems depending upon traffic anomaly [16] [17]. The former primarily develops a DDoS attack characteristics database by gathering various types of attack characteristic information. We can determine whether a network is being attacked by DDoS by analysing present network packet and features database. Expert systems, state transition, model reasoning, and characteristics matching are the primary implementation techniques. The purpose of the latter is primarily to create a traffic model and analyse variations in flow that are abnormal, determining if the traffic is irregular or not in order to identify whether the server has been attacked [18] [19] [20].

Section 2 and 3 reviews extant works and portrays about DDOS attack detection in SDN. Section 4 and 5 described about features and DMO based attack detection in SDN. Section 6 and 7 describes bait process and results.

The contributions are as follows:

1. Using the Mininet emulator, the SDN-specific dataset for both normal and attack flow was generated.
2. DDoS detection takes place using deep max out classifier,

[1,2]School of Computer Science and Engineering,
Vellore Institute of Technology, Chennai, Tamil Nadu 600127, India
ORCID ID: 0000-0002-8319-9371, 0000-0003-2706-2239,
** Corresponding Author Email: karmel.a@vit.ac.in*

wherein, weights are optimized using SMI-CA model.

2. Related Work

Nagarathna et al. [1] focused to prevent DDoS attacks in 2020 that were brought on by malevolent wireless IoT servers. To lessen assault on IoT servers, our security scheme deployed cloud and SDN concept. Also, we have suggested LEDEM, which identifies DDoS and mitigates it. We emulated topology and evaluated LEDEM in the test bed, and then we compared the outcomes to cutting-edge approaches. Our increased DDoS attack detection accuracy rate was 96.28 percent.

A modularized framework that enables recognition and prevention of LR-DDoS threats in SDN environments was presented by Pérez et al. in 2020 [2]. We specifically use six ML models to train the IDS in our design utilising CIC DoS database to assess their efficacy. The threat detection system in our experimental design mitigates any threats that have already been picked up by the IDS system. This shows how effective our architecture is in detecting and thwarting LR-DDoS assaults.

Dong et al. [3] suggested two techniques in 2020 for detecting DDoS attack in SDN. One approach uses the DDoS attack's intensity to determine its level. The alternative technique finds the DDoS assault using the enhanced KNN scheme depending upon ML. Theoretical analytical findings and experimental findings from datasets demonstrated that the suggested techniques can well identify the DDoS attack distinguished when other techniques.

SVM using KPCA and GA was suggested by Sahoo et al. in 2020[4]. KPCA is utilised in the suggested SVM model to decrease the dimension of the feature vectors, while GA is employed to optimise various SVM parameters. An enhanced kernel function is suggested in order to lessen the noise brought on by feature discrepancies. According on the experimental findings, the suggested model gives more precise classification with greater generalisation when compared to single-SVM.

A DCNN ensemble approach for effective DDoS attack detection in SDNs was suggested by Haider et al. in 2020[5]. A conventional Flow oriented dataset is used to assess the proposed system against predetermined standards. Improved accuracy is shown in comparison to current relevant detection methods.

3. Explanation on DDoS Attack Recognition in SDN System

3.1. Architecture

Figure 1 shows the picture of proposed detection model. The adopted DDoS attack recognition in SDN comprises following steps.

- Primarily, "features like flow-based features and statistical features (mean, median, standard deviation, variance, skewness and kurtosis) are derived".
- Further, detection takes place via Deep Max out classifier.
- To enhance the performance of detection, the weights of Deep Maxout classifier are chosen via SMI-CA model.
- Once the presence of attacks is determined, Bait oriented mitigation is used to mitigate the corresponding attacker from the network.

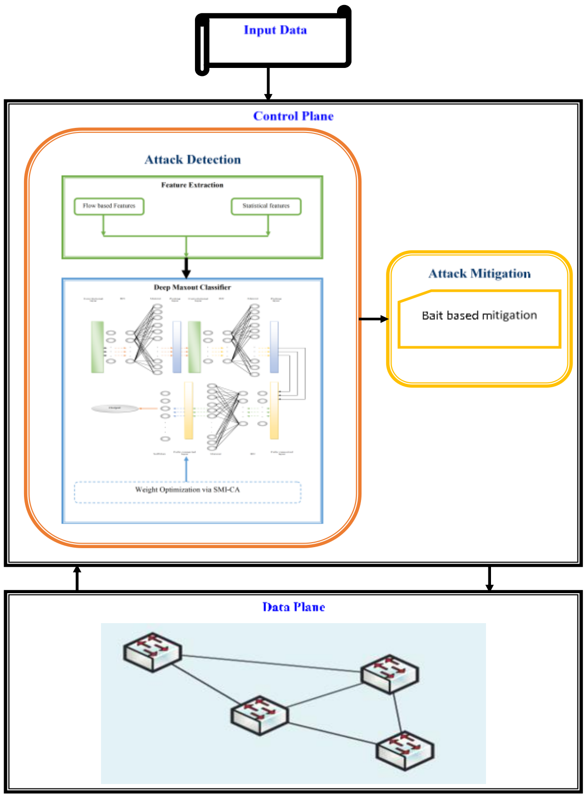


Fig.1 Demonstration of adopted DDoS attack detection in SDN

4. Feature Extraction: Statistical and Flow Based Features

The considered features are on detecting the attacks is as follows:

- Flow based features
- Statistical features

4.1 Flow based Features

These include "source-destination IP addresses and ports as well as protocol types, in addition to the transactional features that includes flow data like data lengths. For DDoS attacks, the features namely, Source IP address (srcip), Source port number (port), Destination IP address (dstip), Destination port number (dsport), Protocol type (proto) and Last time of connection (ltime)" are derived.

4.2 Statistical Features

These include mean, median, variance; kurtosis, standard deviation and skewness are derived.

4.2.1 Skewness [21]: Skewness is a distortion or lack of symmetry which deviates from the symmetric curve or normal distortion in a set of data. The data is mentioned as skewness, when the curve is shifted to right or left of the centre point. It is modelled in Eq. (1).

$$skewness = \frac{\sum_{i=1}^{k} (Z_i - \mu)^3 / k}{std.dev^3} \tag{1}$$

In Eq. (1), $Z_i = Z_1, Z_2, ..., Z_k$, μ →mean value and k → data point count.

4.2.2 Kurtosis [21]: Kurtosis is a statistical measure of the tailedness of a distribution. Excess kurtosis is the tailedness of a distribution relative to a normal distribution. It is modelled in Eq. (2).

$$kurtosis = \frac{\sum_{i=1}^{k} (Z_i - \bar{Z})^4 / k}{std.dev^4} \tag{2}$$

The derived features are DMO for choosing the best features.

5. Optimized DMO Based Attack Detection in SDN

This work exploits DMO for attack detection in SDN.

5.1 Optimized DMO

In the "max out" layer, the activation function acts as the layer's maximum input. Any function may be approximated by an MLP with two maxima out units. They offer a lot of explanations for why max out is effective, but the one that follows is the most important [22].

The dropout model averaging method trains a random sub network for all iterations, and the weights of all the networks are then summed. Since it is challenging to accurately average the weights, an approximation is utilised. For a linear network, this approximation is accurate. In max out, the input to a layer is not dropped. Here, the weights are tuned optimally by a new SMI-CA algorithm during the training process. The algorithm introduced is given in the subsequent section. The model results the classification outcomes. As per the datasets used, the categorization takes place. For Dataset 1: Benign, LDAP, MSSQL, NetBIOS, UDP gets classified. For Dataset 2, the presence or absence of attacks will be determined.

5.2 Proposed SMI-CA Model for weight Tuning

The objective is to reduce the error as in Eq. (3). The DMO weights are chosen with SMI-CA scheme optimally as in Fig. 2.

$$Obj = \min(error) \tag{3}$$

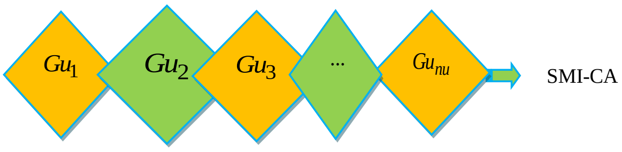


Fig.2. Solution Encoding

The new CA [23] paradigm provided varied benefits; however, special modifications are needed as a result, and SMI-CA is set up. "Usually, conservative optimization techniques are capable of self-enhancements" [24] [25] [26] [27] [28] [29] [30] [31].

In CA, the leaders are measured as % of whole presumed coots "populations, M_{po}" and the residual are followers of coots. The follower's positions (pos_{ct0}) and leaders (pos_{lea}) are initialized randomly as in Eq. (4), and (5), where, ub and lb signifies upper and lower limits. In SMI-CA, the random integers ra_{ct}, ra_{lea} and r are generated chaotically using sine map.

$$pos_{ct0} = ra_{ct}.(ub - lb) + lb \tag{4}$$

$$pos_{lea} = ra_{lea}.(ub - lb) + lb \tag{5}$$

The fitness of coot's followers fit_{coot} is computed as $of(f_{ob})$ in Eq. (6). The best global score gbe_{sco} and its position gbe_{pos} is in Eq. (7). $M_{lea} \rightarrow$ coot leader count = % of M_{po} and $M_{coot} \rightarrow$ coot follower count = M_{po} - M_{lea}.

$$fit_{coot}(1,i) = f_{ob}(p_{coot}(i)),\ i = 1\ to\ M_{coot}) \tag{6}$$

$$\text{If } \left.\begin{array}{l} gbe_{sco} > fit_{coot}(1,i) \\ gbe_{sco} = fit_{coot}(1,i) \\ gbe_{pos} = pos_{coot}(i) \end{array}\right\} \tag{7}$$

Also, the fitness of all coots' leaders via OF is in Eq. (8). The gbe_{sco} & gbe_{pos} is shown in (9).

$$fit_{lea}(1,i) = f_{ob}(p_{lea}(i)),\ i \in M_{lea}) \tag{8}$$

$$\text{If } \left.\begin{array}{l} gbe_{sco} > fit_{lea}(1,i) \\ gbe_{sco} = fit_{lea}(1,i) \\ gbe_{pos} = pos_{lea}(i) \end{array}\right\} \tag{9}$$

Every coot's follower is allocated to coot's leaders beginning from iteration 2 to maximal iterations ($t_{\max}$) as in Eq. (10) and (11). Conventionally, followers' position is updated as shown in Eq. (11). As per SMI-CA, followers' position is updated based upon Brownian motion (BM) as shown in Eq. (12). Also, arithmetic crossover is carried out to make sure on better rate of convergence.

$$r = 1 + 2.r_{coot} \tag{10}$$

$$pos_{coot}(i) = 2.r_{coot}.\cos(2\pi r)\,[pos_{lea}(k) - pos_{coot}(i)] + pos_{lea}(k) \tag{11}$$

$$pos_{coot}(i) = 2.r_{coot}.\cos(2\pi r)\,[pos_{lea}(k) - pos_{coot}(i)] + pos_{lea}(k) + BM \tag{12}$$

Here r_{coot} and r_{lea} implies randomly generated coot's followers and leaders.

If the follower fitness > corresponding leader, the follower and leader interchange their position as in (13).

$$\text{If } \left.\begin{array}{l} fit_{coot}(1,i) < fit_{lea}(1,k), then \\ fit_{lea}(1,k) = fit_{coot}(1,i) \& \\ pos_{lea}(k) = pos_{coot}(i) \end{array}\right\} \tag{13}$$

The leader's positions are improved as in Eq. (14), and (15). The gbe_{sco} and gbe_{pos} are in (16), $it(L) \rightarrow$ iteration count L.

$$\left.\begin{array}{l} B = 2 - (it(L)/it_{\max}) \\ r = 1 + 2.r_{lea} \end{array}\right\} \tag{14}$$

$$pos_{lea} = B.r_{lea}2.r_{coot}.\cos(2\pi r)\left[gbe_{pos} - pos_{lea}(i)\right] + gbe_{pos} \tag{15}$$

$$\text{If } \left.\begin{array}{l} gbe_{sco} > fit_{lea}(1,i)\ then \\ fit_{lea}(1,k) = gbe_{sco} \\ pos_{lea}(i) = gbe_{pos} \end{array}\right\} \tag{16}$$

Algorithm: SMI-CA

Start

Initialize the coot parameters M_{po} and $t_{\max}$

Initialize the COOT's followers pos_{coot} and leaders' position pos_{lea} via Eq. (4) and Eq. (5) → Novelty (sine map)

Evaluate the fitness of each COOT's follower via Eq. (6)

Update the best position pos_{coot} and its best solution via Eq. (7)

Evaluate the fitness of each COOT's leader via Eq. (8)

Update the best position pos_{lea} and its best solution via Eq. (9)

For $t = 2 : t_{\max}$

Update the position of COOT's follower pos_{coot} via Eq. (12) → Novelty (Brownian Motion)

Compare fit_{coot} and fit_{coot}

If $fit_{coot} < fit_{lea}$

$fit_{lea} = fit_{coot}$

else

No updating in fit_{coot} and fit_{lea}

end

Update the position of COOT'S leader pos_{lea} via Eq. (15)

Evaluate the fitness of new leader gbe_{sco}

For $i = 1 : M_{lea}$

If $fit_{lea} < gbe_{sco}$

$fit_{lea} = gbe_{sco}$

$pos_{lea} = gbe_{pos}$

else

No updating in gbe_{sco} and pos_{lea}

end

end

$t = t + 1$

end

Return best solution

End

6. Bait Based Mitigation Process

Once the attacks are determined in the network, it is very important to mitigate it from the network. For this, BAIT based mitigation is followed in this work. As source nodes attempt to broadcast an RREQ (Route Request) to the neighbouring nodes and the neighbouring nodes acknowledge the source node as RREP (Route Reply). Thus, the source node collates its RREQ with RREP to identify the attacking nodes. Let the neighbouring node, chosen at random by the source nodes, be a_n. The source node first sends out a request message $RREQ$ with information like "destination ID as ID_d, source ID as ID_s and path length pl " as in Eq. (17).

$$RREQ = \{ID_s, ID_d, pl\} \tag{17}$$

The pl offer data related to hop count to transmit the requests. Nodes sends feedback after receiving the request as in Eq. (18) that shows the request efficiently arrived at the last node.

$$RREP = \{ID_s, ID_d, pl\} \tag{18}$$

If $RREP$ arrives source, it is evaluated with $RREQ$. As destination and pl is accumulated in $RREQ$, it distinguishes the attacker nodes without difficulty and abolish those nodes

7. Results and Discussion

7.1. Simulation Setup

The developed model was implemented in "Python using two datasets, where dataset 1 is downloaded from [32] and dataset 2 is synthetically generated via simulating SDN in Mininet, and the description is given below". The DMO + SMI-CA was assessed over DMO + TOA, DMO + SSA, DMO + SMO and DMO + CA on miscellaneous metrics. In addition, measurement was done with SVM, DBN, CNN and RNN.

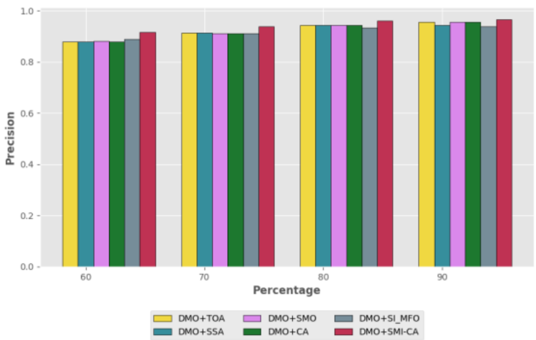
Dataset description: CICDDOS 2019: Distributed Denial of Service (DDoS) attack is a menace to network security that aims at exhausting the target networks with malicious traffic. Although many statistical methods have been designed for DDoS attack detection, designing a real-time detector with low computational overhead is still one of the main concerns. On the other hand, the evaluation of new detection algorithms and techniques heavily relies on the existence of well-designed datasets.

Mininet Environment Specifications: A DELL Inc. Inspiron15 5000 computer with the following specs was utilised for all testing and experiments: Intel Core (TM) i5-10th Gen processor clocked at 1.00GHz, 8 GB of RAM, Windows 10 64-bit operating system, and VirtualBox Oracle VM version 6.0.18. On this machine, under the control of VirtualBox, is installed the guest operating system: MININET Emulator version 2.3.1b1 on Linux operating system Ubuntu 14.0432bits with 4096 MB of RAM and RYU Controller.

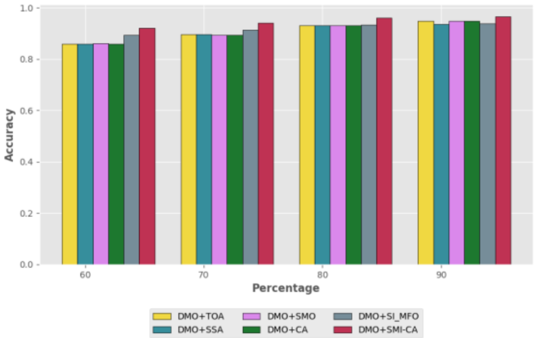
SDN Simulation Dataset: This dataset is manually generated by simulating SDN in Mininet. The customized topology was created with four hosts, three switches, two servers and one RYU controller. The attributes included are flow duration, ip_proto, srcport, byte count, packet count, type.

7.2 Performance Study

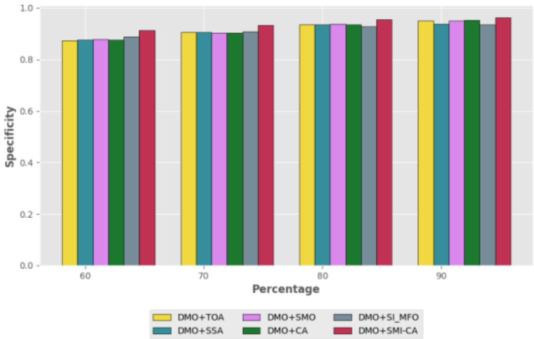
The inspection on DMO + SMI-CA is done over existing optimizing schemes such as DMO + TOA, DMO + SSA, DMO + SMO, DMO + CA and DMO+SI-MFO on disparate metrics. Consequently, the inspection on DMO + SMI-CA is done over existing classifiers like SVM, DBN, CNN and RNN. The assessment of DMO + SMI-CA done over DMO + TOA, DMO + SSA, DMO + SMO, DMO + CA and DMO+SI-MFO models is exposed in Fig. 3- 4. The analysis on FPR and FNR is shown in Fig. 5 and 6 for datasets 1 and 2, whereas, MCC, NPV and F-measure is shown in Fig. 7 and 8 for datasets 1 and 2. The MCC, NPV and F-measures are high for all LPs than evaluated methods, particularly; it is high at 90th LP. The FPR and FNR metrics are low for DMO + SMI-CA technique. Table 1 described the estimation of DMO + SMI-CA over conventional SVM, DBN, CNN and RNN. Here, DMO + SMI-CA was found to have best results at 90th LP over other LPs for dataset 1. For dataset 2, a high specificity is gained at 90th LP. The precision is high at 90th LP. In Table 1, DMO + SMI-CA has gained best specificity of 0.93. Furthermore, DBN was established to be most excellent next to DMO + SMI-CA. Thus, DMO + SMI-CA is confirmed over DMO + TOA, DMO + SSA, DMO + SMO and DMO + CA, DMO+SI-MFO, SVM, DBN, CNN and RNN.



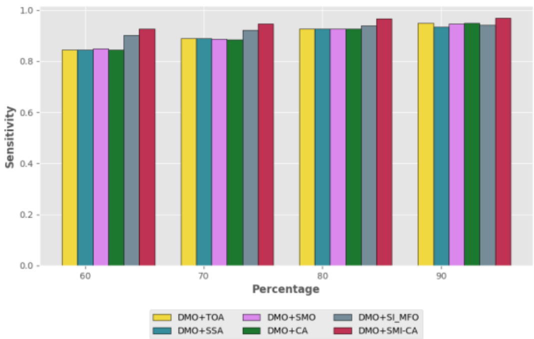
(a)



(b)

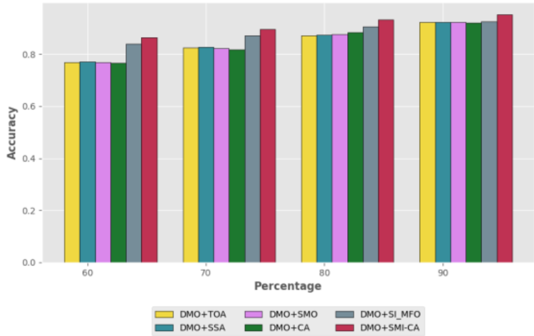


(c)

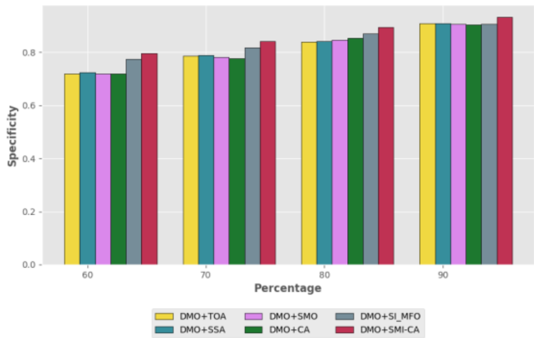


(d)

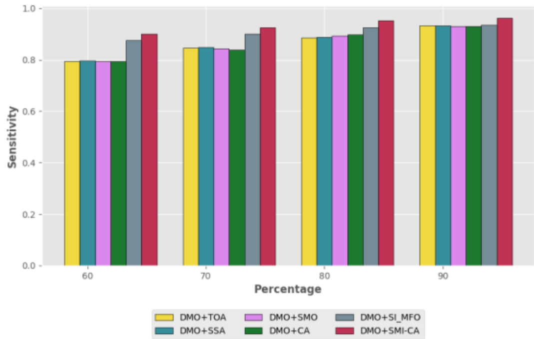
Fig. 3. Analysis via DMO + SMI-CA over other schemes for "(a) Precision (b) Accuracy (c) Specificity and (d) Sensitivity" for dataset 1



(b)

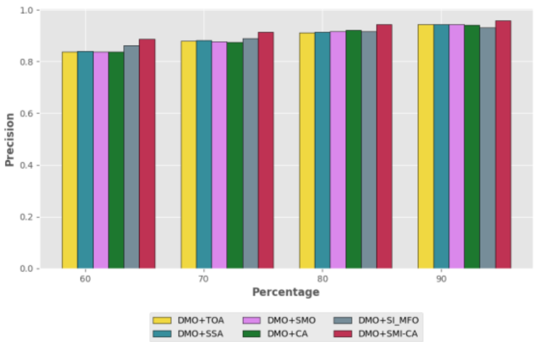


(c)

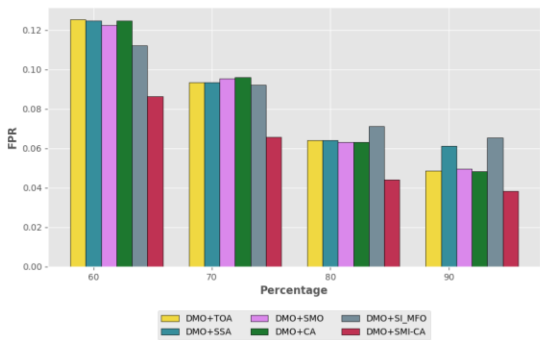


(d)

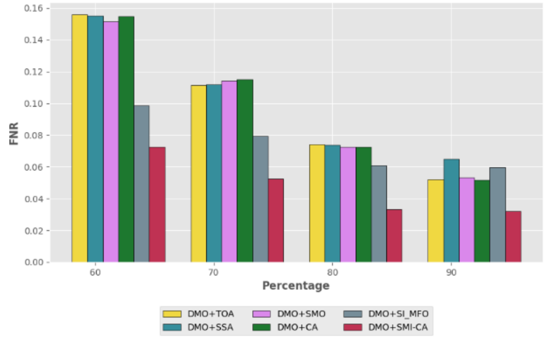
Fig. 4. Analysis via DMO + SMI-CA over other schemes for "(a) Precision (b) Accuracy (c) Specificity and (d) Sensitivity" for dataset 2



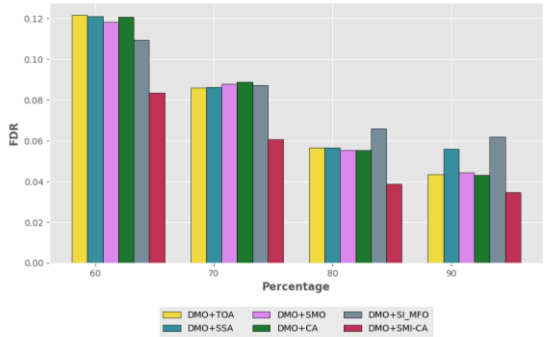
(a)



(a)

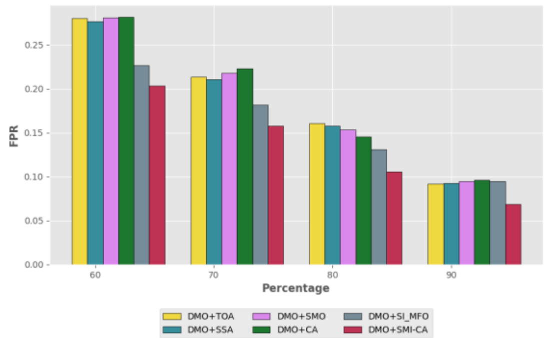


(b)

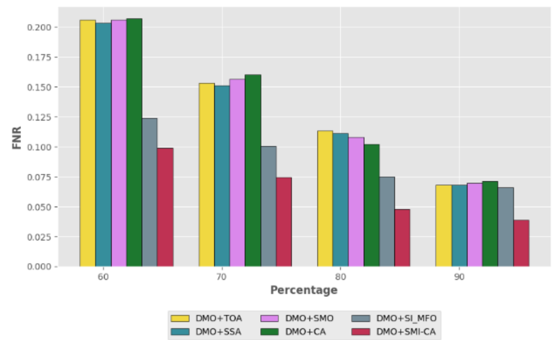


(c)

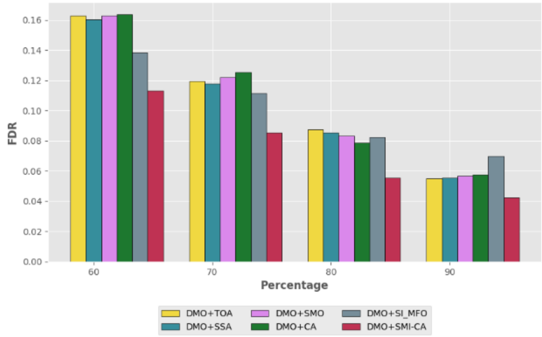
Fig.5. Analysis via DMO + SMI-CA over other schemes for "(a) FPR (b) FNR and (c) FDR" for dataset 1



(a)

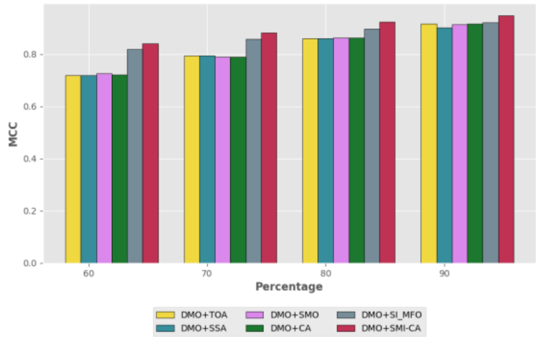


(b)

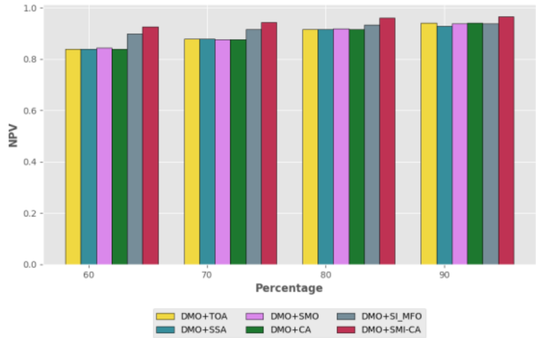


(c)

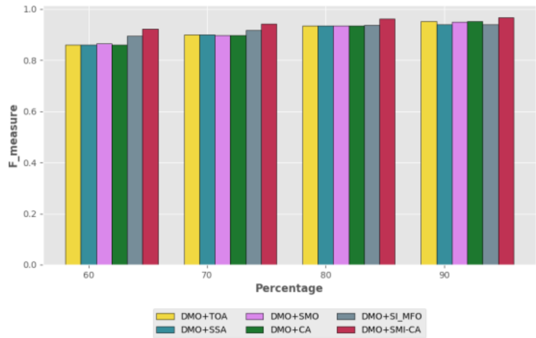
Fig. 6. Analysis via DMO + SMI-CA over other schemes for "(a) FPR (b) FNR and (c) FDR" for dataset 2



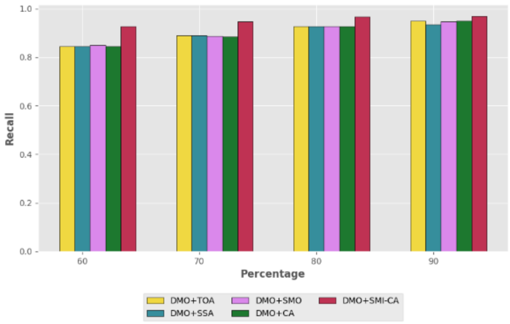
(a)



(b)

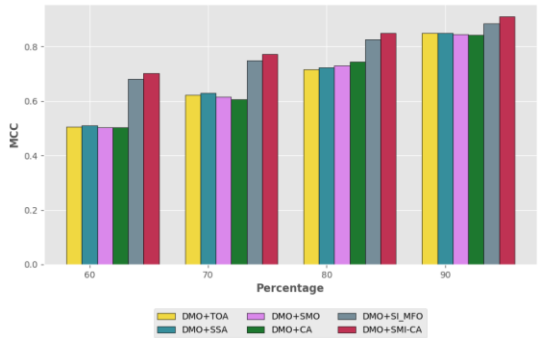


(c)

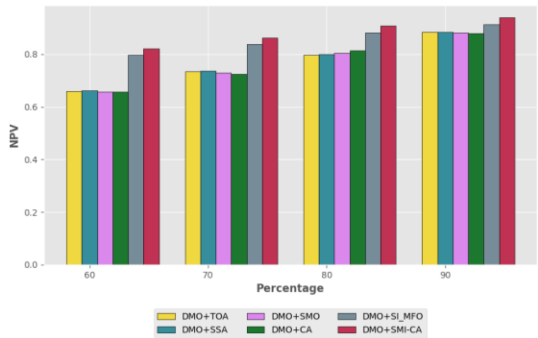


(d)

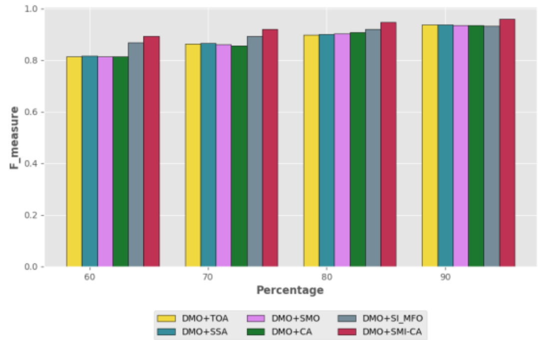
Fig.7. Analysis via DMO + SMI-CA over other schemes for "(a) MCC (b) NPV (c) F-measure and (d) Recall" for dataset 1



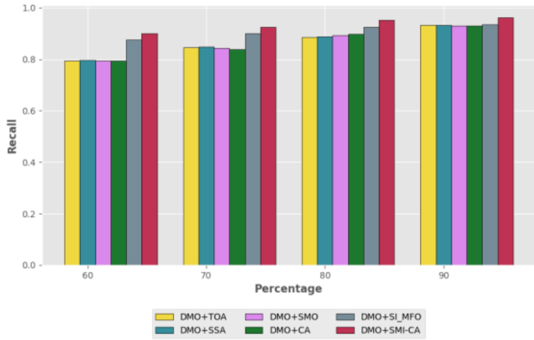
(a)



(b)



(c)



(d)

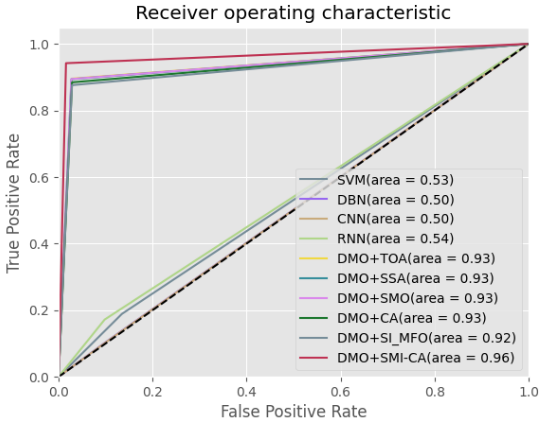
Fig.8. Analysis via DMO + SMI-CA over other schemes for "(a) MCC (b) NPV (c) F-measure and (d) Recall" for dataset 2

Table 1. Analysis via DMO + SMI-CA over other classifier schemes

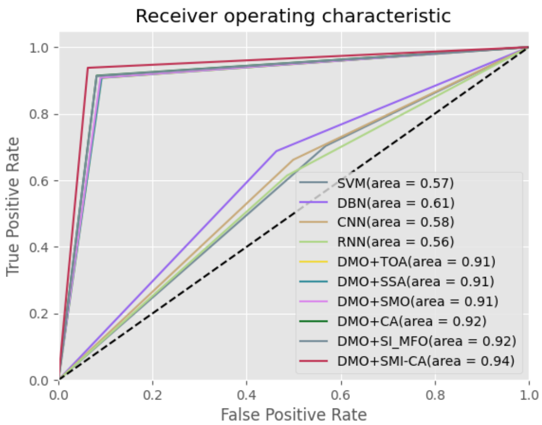
Metrics	SVM	DBN	CNN	RNN	DMO + SMI-CA
Dataset 1					
Accuracy	0.851138	0.927594	0.901549	0.77875	0.941035
Sensitivity	0.84	0.919712	0.913325	0.702405	0.947489
Specificity	0.859813	0.935975	0.89038	0.828304	0.934092
Precision	0.823529	0.938551	0.887678	0.726425	0.939268
F_measure	0.831683	0.929036	0.900319	0.714213	0.943361
MCC	0.698379	0.855327	0.803425	0.534006	0.88191
NPV	0.873418	0.916417	0.915468	0.8109	0.942971
Recall	0.84	0.919712	0.913325	0.702405	0.947489
FDR	0.176471	0.061449	0.112322	0.273575	0.060732
FPR	0.140187	0.064025	0.10962	0.171696	0.065908
FNR	0.16	0.080288	0.086675	0.297595	0.052511
Dataset 2					
Accuracy	0.789062	0.844362	0.844956	0.864	0.896131
Sensitivity	0.841254	0.863562	0.894487	0.888889	0.925716
Specificity	0.701962	0.808413	0.760711	0.818182	0.842292
Precision	0.824886	0.894058	0.864095	0.9	0.914396
F_measure	0.83299	0.878546	0.879028	0.89441	0.920021
MCC	0.54704	0.662898	0.664143	0.703526	0.772044
NPV	0.726004	0.759885	0.809114	0.8	0.861706
Recall	0.841254	0.863562	0.894487	0.888889	0.925716
FDR	0.175114	0.105942	0.135905	0.1	0.085604
FPR	0.298038	0.191587	0.239289	0.181818	0.157708
FNR	0.158746	0.136438	0.105513	0.111111	0.074284

7.3 ROC Analysis

Fig. 9(a) and (b) shows the ROC analysis done via deployed DMO + TOA, DMO + SSA, DMO + SMO, DMO + CA, and DMO+SI-MFO SVM, DBN, CNN and RNN. The ROC is analysed for TPR and FPR. For both datasets, a high ROC of 1.0 is obtained for DMO + SMI-CA. Also, a high area of 0.94 is obtained for DMO + SMI-CA for dataset 2. Thus, with increase in FPR, a high TPR is attained for developed scheme



(a)

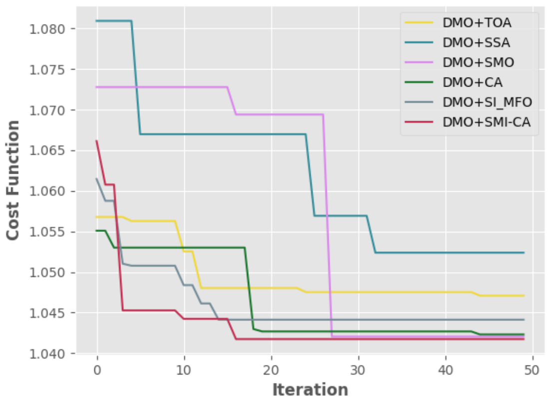


(b)

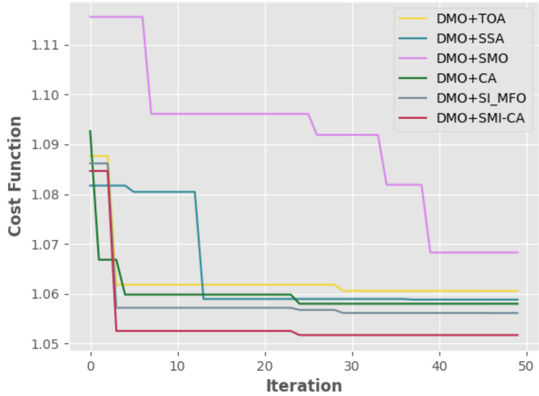
Fig.9. ROC curve for varied methods using dataset (a) 1 and (b) 2

7.4 Convergence Analysis

The convergence of SMI-CA scheme over DMO + TOA, DMO + SSA, DMO + SMO, DMO + CA and DMO+SI-MFO for varied iterations (0-50) is shown in Fig. 10 (a) and (b) for both datasets. The cost has to be less as attained by SMI-CA from 16th to 50th iteration. From Fig. 10 (b), a lesser cost of 1.053 is gained by SMI-CA over DMO + TOA, DMO + SSA, DMO + SMO, DMO + CA and DMO+SI-MFO. Thus, enhanced results are gained using SMI-CA scheme.



(a)



(b)

Fig. 10. Convergence analysis of SMI-CA over others using dataset (a) 1 and (b) 2

7.5. Statistical Analysis

The statistical analysis of developed DMO+SMI-CA with existing methods is illustrated in Table 2 respectively. And, on noticing the mean of the proposed model DMO+SMI-CA is 1.043777 where the existing DMO+TOA=1.049583, DMO+SSA=1.061707, DMO+SMO=1.057901 and DMO+CA=1.046439 for dataset 1. And, it is observed that the developed model attains best mean value than the traditional methods. Moreover, on observing Table 2 for dataset 2 the developed model holds 1.054037 mean which is 0.82%, 0.99%, 3.35% and 0.55% superior than DMO+TOA, DMO+SSA, DMO+SMO, and DMO+CA respectively.

Table 2: Statistical Analysis via DMO + SMI-CA over other schemes

Metrics	Standard deviation	Mean	Median	Maximum	Minimum
Dataset 1					
DMO+TOA	0.003627	1.049583	1.047525	1.056776	1.047086
DMO+SSA	0.00918	1.061707	1.061945	1.08093	1.052379
DMO+SMO	0.014826	1.057901	1.069398	1.072776	1.042054
DMO+CA	0.005164	1.046439	1.042677	1.055081	1.042308
DMO+SI_MFO	0.004193	1.046243	1.044124	1.061449	1.044124
DMO+SMI-CA	0.00502	1.043777	1.041734	1.066122	1.041734
Dataset 2					
DMO+TOA	0.006362	1.062857	1.061828	1.087671	1.060562
DMO+SSA	0.009762	1.064636	1.058954	1.081717	1.058832
DMO+SMO	0.014855	1.09061	1.096104	1.115564	1.068276
DMO+CA	0.005171	1.059962	1.058001	1.092643	1.058001
DMO+SI_MFO	0.007089	1.058447	1.05677	1.086159	1.056137
DMO+SMI-CA	0.007823	1.054037	1.051712	1.084647	1.051712

8. Conclusion and Future Work

This paper suggested a new DDOS attack recognition model in SDN, where, primarily, “features like flow based and statistical features (mean, median, standard deviation, variance, skewness and kurtosis)” were derived. Further, detection was done using Deep Max out classifier, whose weights were chosen via SMI-CA model. If any attack was found, Bait oriented mitigation was made for relieving from attacks. Here, DMO + SMI-CA was found to have best results at 90th LP over other LPs for dataset 1. For dataset 2, a high specificity is gained at 90th LP. The accuracy was elevated at 90th LP. Also, DMO + SMI-CA has gained best specificity of 0.93. Furthermore, DBN was established to be most excellent next to DMO + SMI-CA.

For the future work, in order to increase the performance of Deep Learning classifiers against attacks from the CIC attack dataset as well as with simulated datasets other than DDoS attack, the suggested work will be extended to include newer hybrid metaheuristic optimization techniques.

References

[1] 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, April 2020, doi: 10.1109/JIOT.2020.2973176.

[2] J. A. Pérez-Díaz, I. A. Valdovinos, K. -K. R. Choo and D. Zhu, "A Flexible SDN-Based Architecture for Identifying and Mitigating Low-Rate DDoS Attacks Using Machine Learning," IEEE Access, vol. 8, pp. 155859-155872, 2020, doi: 10.1109/ACCESS.2020.3019330.

[3] S. Dong and M. Sarem, "DDoS Attack Detection Method Based on Improved KNN With the Degree of DDoS Attack in Software-Defined Networks," in IEEE Access, vol. 8, pp. 5039-5048, 2020, doi: 10.1109/ACCESS.2019.2963077.

[4] Krishna, P. R. ., and P. . Rajarajeswari. “EapGAFS: Microarray Dataset for Ensemble Classification for Diseases Prediction”. International Journal on Recent and Innovation Trends in Computing and Communication, vol. 10, no. 8, Aug. 2022, pp. 01-15, doi:10.17762/ijritcc.v10i8.5664.

[5] K. S. Sahoo et al., "An Evolutionary SVM Model for DDOS Attack Detection in Software Defined Networks," in IEEE Access, vol. 8, pp. 132502-132513, 2020, doi: 10.1109/ACCESS.2020.3009733.

[6] S. Haider et al., "A Deep CNN Ensemble Framework for Efficient DDoS Attack Detection in Software Defined Networks," in IEEE Access, vol. 8, pp. 53972-53983, 2020, doi: 10.1109/ACCESS.2020.2976908.

[7] Y. Yu, L. Guo, Y. Liu, J. Zheng and Y. Zong, "An Efficient SDN-Based DDoS Attack Detection and Rapid Response Platform in Vehicular Networks," in IEEE Access, vol. 6, pp. 44570-44579, 2018, doi: 10.1109/ACCESS.2018.2854567.

[8] M. A. Aladaileh, M. Anbar, I. H. Hasbullah, Y. -W. Chong and Y. K. Sanjalawe, “Detection Techniques of Distributed Denial of Service Attacks on Software-Defined Networking Controller–A Review," in IEEE Access, vol. 8, pp. 143985-143995, 2020, doi: 10.1109/ACCESS.2020.3013998.

[9] K. S. Sahoo et al., "An Evolutionary SVM Model for DDOS Attack Detection in Software Defined Networks," in IEEE Access, vol. 8, pp. 132502-132513, 2020, doi: 10.1109/ACCESS.2020.3009733.

[10] Z. Abou El Houda, L. Khoukhi and A. Senhaji Hafid, "Bringing Intelligence to Software Defined Networks: Mitigating DDoS Attacks," in IEEE Transactions on Network and Service Management, vol. 17, no. 4, pp. 2523-2535, Dec. 2020, doi: 10.1109/TNSM.2020.3014870.

[11] M. P. Novaes, L. F. Carvalho, J. Lloret and M. L. Proença, "Long Short-Term Memory and Fuzzy Logic for Anomaly Detection and Mitigation in Software-Defined Network Environment," in IEEE Access, vol. 8, pp. 83765-83781, 2020, doi: 10.1109/ACCESS.2020.2992044.

[12] B. Wang, Y. Sun and X. Xu, "A Scalable and Energy-Efficient Anomaly Detection Scheme in Wireless SDN-Based mMTC Networks for IoT," in IEEE Internet of Things Journal, vol. 8, no. 3, pp. 1388-1405, 1 Feb.1, 2021, doi: 10.1109/JIOT.2020.3011521.

[13] T. V. Phan, T. G. Nguyen, N. -N. Dao, T. T. Huong, N. H. Thanh and T. Bauschert, "DeepGuard: Efficient Anomaly Detection in SDN With Fine-Grained Traffic Flow Monitoring," in IEEE Transactions on Network and Service Management, vol. 17, no. 3, pp. 1349-1362, Sept. 2020, doi: 10.1109/TNSM.2020.3004415.

[14] Y. Xu, H. Sun, F. Xiang and Z. Sun, "Efficient DDoS Detection Based on K-FKNN in Software Defined Networks," in IEEE Access, vol. 7, pp. 160536-160545, 2019, doi: 10.1109/ACCESS.2019.2950945.

[15] Ghazaly, N. M. . (2022). Data Catalogue Approaches, Implementation and Adoption: A Study of Purpose of Data Catalogue. International Journal on Future Revolution in Computer Science &Amp; Communication Engineering, 8(1), 01–04. https://doi.org/10.17762/ijfrcsce.v8i1.2063

[16] B. Wang, Y. Sun and X. Xu, "Loose Game Theory Based Anomaly Detection Scheme for SDN-Based mMTC Services," in IEEE Access, vol. 7, pp. 139350-139357, 2019, doi: 10.1109/ACCESS.2019.2943056.

[17] Bawany, N.Z., Shamsi, J.A. & Salah, K. DDoS Attack Detection and Mitigation Using SDN: Methods, Practices, and Solutions. Arab J Sci Eng 42, 425–441 (2017).

[18] Wani, A., Revathi, S. DDoS Detection and Alleviation in IoT using SDN (SDIoT-DDoS-DA). J. Inst. Eng. India Ser. B 101, 117–128 (2020). https://doi.org/10.1007/s40031-020-00442-z

[19] Bhushan, K., Gupta, B.B. Distributed denial of service (DDoS) attack mitigation in software defined network (SDN)-based cloud computing environment. J Ambient Intell Human Comput 10, 1985–1997 (2019). https://doi.org/10.1007/s12652-018-0800-9

[20] Harikrishna, P., Amuthan, A. SDN-based DDoS Attack Mitigation Scheme using Convolution Recursively Enhanced Self Organizing Maps. Sādhanā 45, 104 (2020). https://doi.org/10.1007/s12046-020-01353-x

[21] Sahoo, K.S., Panda, S.K., Sahoo, S. et al. Toward secure software-defined networks against distributed denial of service attack. J Supercomput 75, 4829–4874 (2019). https://doi.org/10.1007/s11227-019-02767-z

[22] Mousavi, S.M., St-Hilaire, M. Early Detection of DDoS Attacks Against Software Defined Network Controllers. J Netw Syst Manage 26, 573–591 (2018). https://doi.org/10.1007/s10922-017-9432-1

[23] https://www.itl.nist.gov/div898/handbook/eda/section3/eda35b.htm#:~:text=Skewness%20is%20a%20measure%20of,relative%20to%20a%20normal%20distribution.

[24] https://www.kaggle.com/code/accountstatus/maxout-network-vs-normal-cnn/notebook

[25] IrajNaruei, FarshidKeynia, "A new optimization method based on COOT bird natural life model", Expert Systems With Applications, vol. 183, 2021.

[26] B. R. Rajakumar, "Impact of Static and Adaptive Mutation Techniques on Genetic Algorithm", International Journal of Hybrid Intelligent Systems, vol. 10, no. 1, pp. 11-22, 2013, DOI: 10.3233/HIS-120161.

[27] B. R. Rajakumar, "Static and Adaptive Mutation Techniques for Genetic algorithm: A Systematic Comparative Analysis", International Journal of Computational Science and Engineering, vol. 8, no. 2, pp. 180-193, 2013, DOI: 10.1504/IJCSE.2013.053087.

[28] S. M. Swamy, B. R. Rajakumar and I. R. Valarmathi, “Design of Hybrid Wind and Photovoltaic Power System using

Opposition-based Genetic Algorithm with Cauchy Mutation", IET Chennai Fourth International Conference on Sustainable Energy and Intelligent Systems (SEISCON 2013), Chennai, India, Dec. 2013, DOI: 10.1049/ic.2013.0361

[29] Paithane, P. M., & Kakarwal, D. (2022). Automatic Pancreas Segmentation using A Novel Modified Semantic Deep Learning Bottom-Up Approach. International Journal of Intelligent Systems and Applications in Engineering, 10(1), 98–104. https://doi.org/10.18201/ijisae.2022.272

[30] Aloysius George and B. R. Rajakumar, "APOGA: An Adaptive Population Pool Size based Genetic Algorithm", AASRI Procedia - 2013 AASRI Conference on Intelligent Systems and Control (ISC 2013), vol. 4, pp. 288-296, 2013, DOI: https://doi.org/10.1016/j.aasri.2013.10.043.

[31] Ahmed Cherif Megri, Sameer Hamoush, Ismail Zayd Megri, Yao Yu. (2021). Advanced Manufacturing Online STEM Education Pipeline for Early-College and High School Students. Journal of Online Engineering Education, 12(2), 01–06. Retrieved from http://onlineengineeringeducation.com/index.php/joee/article/view/47

[32] B. R. Rajakumar and Aloysius George, "A New Adaptive Mutation Technique for Genetic Algorithm", In proceedings of IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), pp. 1-7, Dec 18-20, Coimbatore, India, 2012, DOI: 10.1109/ICCIC.2012.6510293.

[33] Mukund B. Wagh and Dr. Gomathi N, "Improved GWO-CS Algorithm-Based Optimal Routing Strategy in VANET", Journal of Networking and Communication Systems, Vol.2,No.1, pp.34-42,2019.

[34] Sadashiv Halbhavi B,Kodad S F,Ambekar S K,Manjunath D, "Enhanced Invasive Weed Optimization Algorithm with Chaos Theory for Weightage based Combined Economic Emission Dispatch", Journal of Computational Mechanics, Power System and Control, Vol.2,No.3, pp.19-27,2019.

[35] Amolkumar Narayan Jadhav,Gomathi N, "DIGWO: Hybridization of Dragonfly Algorithm with Improved Grey Wolf Optimization Algorithm for Data Clustering", Multimedia Research, Vol.2,No.3, pp.1-11,2019.
https://www.unb.ca/cic/datasets/ddos-2019.html