Hybrid Deep Learning Model for Detecting DDoS Attacks in IoT Networks

Jyothsna Veeramreddy[1*], Chaithanya Kumar Reddy Vardhireddy[2], Hemasree Thangella[3], Kartheek Sarangula[4], Roshini Tamidilapati[5], Bhasha Pydala[6]

[1] Associate Professor and Associate Dean, Dept. of Data Science, Mohan Babu University (Erstwhile Sree Vidyanikethan Engineering College), Tirupati-517102, A.P. India.
[2,3,4,5] UG Scholar, Dept. of Information Technology, Mohan Babu University (Erstwhile Sree Vidyanikethan Engineering College), Tirupati-517102, A.P. India.
[6]Assistant Professor, Dept. of Data Science, Mohan Babu University (Erstwhile Sree Vidyanikethan Engineering College), Tirupati-517 102, A.P. India.

jyothsna1684@gmail.com[1*] chaithuchaithu4u@gmail.com[2]
themasree23@gmail.com[3] sarangulakarthik@gmail.com[4]
roshiniroyal63@gmail.com[5] basha.chanti@gmail.com[6]

Abstract. As the number of internet connected devices has surpassed tens of billions, the era of the "Internet-of-Things" (IoT) is here. These days, a vast array of products seamlessly integrate the internet, from small devices like smartwatches to more intricate systems like smart grids, smart transit networks, and smart cities. Apart from offering several advantages for the way of life, this integration enables a significant amount of routine tasks to be automated Yet, when a gadget is online, it opensit susceptible to hacking attempts by malevolent individuals or other organizations looking to exploit the weaknesses in the device. Growing heterogeneity and diversity of devices increases the frequency of securityflaws and increases the difficulty of patching and resolving them. Attacks by hackers that might affect more devices and a larger variety of targets are now more likely to occur. Cybercriminals are using "Distributed Denial-Of-Service"(DDoS) attacks increasingly to undermine systems. This project aims to create a brand-new intrusion detection system powered by deep learning created for the Internet of Things (IoT), since traditional machine learning is not able to detect these threats in real-world deployment. This technique makes the effective claim to identify and neutralize DDoS attacks inside the particular context of networked devices. The proposed hybrid model combines "Recurrent neural networks"(RNN),"long short-term memory" (LSTM), and "Multilayer perceptron"(MLP) to recognize all sorts of DDoS attacks and their specific subcategories. This dataset --CICDDoS-2019--,compiles with everything which satisfies all intrusion detection dataset requirements, is utilized to evaluate the proposed model.

Keywords: RNN, LSTM, MLP, DDoS, IoT.

1 INTRODUCTION

A array of physically linked items that are integrated with sensors, soft-ware, and other technologies to enable data gathering and exchange is known

K. R. Madhavi et al. (eds.), *Proceedings of the International Conference on Computational Innovations and Emerging Trends (ICCIET 2024)*, Advances in Computer Science Research 112,
https://doi.org/10.2991/978-94-6463-471-6_42

as the "Internet-of-Things" (IoT). These gadgets are found in many different industries, including industrial, transportation, healthcare, and home automation. IoT devices facilitate smooth communication and data interchange, which boosts productivity, ease of use, and creativity across a wide range of industries. The "Internet-of-Things" is a major force defining the forthcoming of technology and commu- nication, whether it is used to optimize energy consumption in homes, monitor health parameters remotely, enhance transportation networks, or streamline in- dustrial operations. Ranging from small gadgets to large-scale equipment, IoT devices interact autonomously via the internet, with Cisco estimating around50 billion connected devices [15]. Security flaws become more frequent as a result of growing device heterogeneity and diversity, and patching and addressing these vulnerabilities becomes more challenging. Hackers are now more likely to launch attacks that might damage more devices and a greater variety of targets. Cyber- criminals are increasingly employing "Distributed Denial-Of-Service" attacksto cause damage to systems.The attacks originated from denial-of-service attacks, in which hackers used a single device to launch an increasingly complex attack. An increasing number of DDoS assaults rely heavily on IoT devices to increase their reach and efficiency. They are typically categorized into volumet- ric attacks as well as attacks at the application layer.TheOSI model's layers three and four are the focus of volumetric or flooding assaults, which deplete network resources and exhausting network bandwidth through protocols such as ICMP, UDP, and TCP-SYN flood [20]. These attacks can serve to mis-direct anddivert attention from simultaneous attacks on websites or applications. DDoS attacks have remained a persistent and challenging network security issue for many years. Despite numerous defense methods proposed in both academic and industrial settings, the threat posed by DDoS attacks continues to be significant and escalates annually [2].

IoT Network Layers: An Internet of Things (IoT) network architecture typically involves multiple layers to handle data processing and management efficiently. These layers often include the fog, edge, and cloud layers:

2 LITERATURE SURVEY

A unique hybrid method called AE-MLP was presented by Wei et al. [2] to recognize and categorize DDoS attacks. Their assessment demonstrated that the AE-MLP approach outperformed other models, achieving impressive performance metrics. Specifically, the AE-MLP model attained a remarkable 98.34 accuracy, along with high recall (98.48), precision (97.91), and F1-score (98.18). These outcomes demonstrate how well the AE-MLP hybrid technique performs in precisely identifying and categorizing DDoS attacks, showcasing its potential for enhancing cybersecurity measures.

In their study, Aydin et al. [1] crafted a system intended to use "Long Short-Term Memory" networks for identification and mitigation of DDoS attack inpublic cloud settings. They employed a technique that identified attack patterns using signatures. They attained an astounding accuracy rate of 99.83 with their LSTM-based model, demonstrating how efficient it is at countering these kinds of attacks. This study highlights how LSTM networks can support cybersecurity defenses, particularly in the area of identification and defense against ddos attacks in cloud computing environments.

In reference, introduces a novel approach to address increasing network attacks due to the exponential growth of internet services, focusing on flow-level features to capture the varied ways in which network traffic behaves.The suggested model makes use of ensemble classifiers with drift detection and distribution similarity, DLMHS, incorporates a “Deep Learning Neural Network” for detecting attacks, achieving validation through experiments on NSL-KDD benchmark datasets and outperforming existing literature models in terms of statistical parameters [19].

In their study, Elsayed et al. [7] endeavor to enhance classification accuracy while minimizing computational complexity through the identification of important elements from the initial dataset. To accomplish this objective, they leverage Using a specially created deep learning architecture based on the “LSTM-Autoencoder”, Information Gain (IG) and Random Forest (RF) feature selection approaches are used. Promising resultsare obtained when their methodology is evaluated, the IG selection approach produced a 99.50 accuracy rate during training, while the random forest selection method produces a 98.76 accuracy rate.

In reference [8], introduces a novel approach for defending against network attacks by focusing on flow-level traffic characteristics rather than packet-level features. Unique flow features are defined and utilized to train an ensemble of classifiers, leveraging meta-heuristic scaling to handle diverse attack behaviors.Through Kolmogorov–Smirnov Test, diverse flow characteristics are identified and used to train ensemble classifiers, effectively capturing varied behaviors.

Priyadarshini and Barik [9] focused on mitigating and identifying DDoS attacks within fog computing environments. They adopted using a deep learning method called “LSTM”, renowned for its effectiveness with sequential data, to devise their model. The evaluation of their method demonstrated a notable accuracy rate of 98.88, emphasizing its efficiency in com- bating DDoS threats within fog computing landscapes. This study represents a significant advancement in bolstering security protocols in fog computing envi- ronments, underscoring the promising capabilities of LSTM-based methodologies in mitigating DDoS vulnerabilities.

In their work [17], the authors proposed a “Oppositional-Crow-Search” Algorithm, which blends the CrSA with Opposition-Based-Learning, is integrated into a mechanism for detecting DoS attacks.The system architecture consists of two main stages: firstly, First, Recurrent Neural Network classifier (RNN) is used for classification, and then OCSA is used for feature selection. Once the key characteristics have been determined through the OCSA method, they are subsequently inputted into the RNN classifier for data categorization.

3 PROPOSED METHODOLOGY

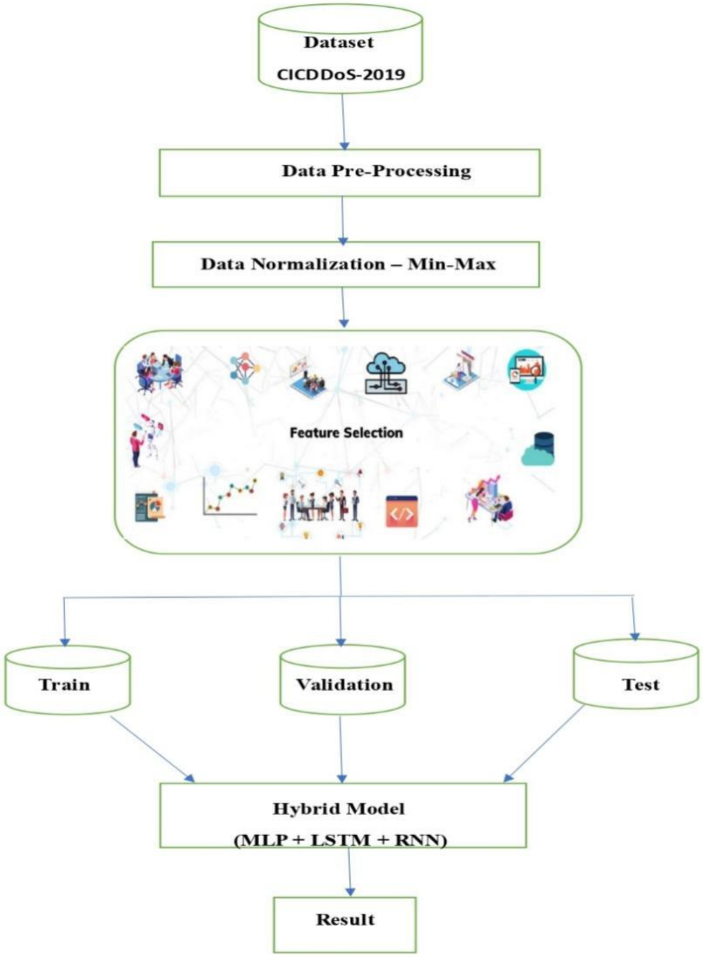


Fig. 1. Architecture of Hybrid Model

3.1 Data Set

In this study, we made use of the CICDDoS-2019 dataset, a meticulously annotated Canadian dataset that includes both benign and malicious attacks for training and testing our proposed system.

This dataset offers a comprehensive spectrum of common DDoS attacks, along with benign traffic, reflecting real-world scenarios captured in PCAPs. Specifically, the CICDDoS-2019 dataset profilesthe general actions of 25 users on several protocols including email, FTP, SSH, HTTP, and HTTPS. It incorporates the CICFlowMeter-V3 tool yielded labeled flows classified by attributes such protocols, attack kinds, source and destination ports, timestamps, and IP addresses. These flows are then stored as the network traffic analysis's conclusions in csv files.

3.2 Data Preprocessing And Normalization

This stage is pivotal and can be time-consuming in data analysis, as it necessitates filtering out extracting the valuable information from the unimportant stuff. Data cleaning and value replacement are done using statistical approaches. considered irrelevant for the experimental analysis. This initial phase of examination is fundamental in converting information into a reliable form. Organizations are able to guarantee that their data is appropriate for analysis thanks to data cleaning, which significantly enhances data quality.

The features are standardized by scaling each feature's greatest value to 1 and lowest value to 0, and proportionally adjusting the remaining values to decimals between 0 and 1. The process involves consolidating the data from Training and testing subsets are created by one table made up of eleven CSV files. Then, negligible attributes like 'Unnamed-0', 'Flow-ID', 'Source-IP','Source-Port', 'Destination-IP', 'Timestamp', Like HTTP, as well as Inbound are not included. Redundant data is then eliminated, and Equation (1) is used to normalize the values of each feature for each instance of feature j.

$$y_i(\mathrm{j}) = y_i(\mathrm{j}) - \frac{\min(y(j))}{\max(y(j)) - \min(y(j))} \qquad \rightarrow (1)$$

3.3 Feature Selection

The Dragonfly Algorithm (DA), introduced in 2015, draws inspiration from the swarming behaviors of dragonflies, emphasizing the exploration and exploitation phases akin to meta-heuristics. In the exploration phase, dragonflies form sub-swarms to traverse various regions, while in the exploitation phase, they navigate in larger swarms along a singular direction. The algorithm combines three fundamental principles of swarming behavior from Reynolds and introduces two additional concepts: attraction to food sources, cohesiveness, separation, alignment, and enemy diversion. These principles emulate both the dynamic and static behaviors observed in dragonflies, which are essential for achieving effective optimization.

3.4 Hybrid Model Classifiers

MLP: A sort of artificial neural network with input, hidden, and output layers is called a Multilayer perceptron.Connected among other layers where, within every stratum neuron process information via weighted connections. Input data is initially introduced into the network, traversing through the hidden layers where nonlinear transformations are applied using activation functions. During the training phase, the MLP adapts its parameters, such as weights and biases, through backpropagation, aiming reduce the difference between expected and real results as much as possible.

Algorithm: MLP Algorithm

Inputs

- The training dataset consisting of features (xTrain) and corresponding labels (yTrain).
- The learning rate parameter denoted by alpha,the number of epochs specified as numEpochs and the configuration of neurons in each hidden layer represented by hiddenLayerSizes.

Steps

1. Initialization of weights and biases can be done randomly or by employing specific initialization techniques.
2. For epoch = 1 to numEpochs:
3. For each training example (x_i, y_i) in the training dataset:
4. Forward propagation
5. Input layer:Set activations of input neurons as x_i
6. Hidden layer: For every layer that is hidden:
7. Determine the total weight of the inputs to every neuron.
8. Apply Activation function to compute neuron
9. Activations
10. Output layer:Calculate weighted sum of inputs to output neurons
11. Apply activation function to compute output
12. Backpropagation
13. Determine the output layer's error by dividing the expected output by the actual output.
14. For each layer (from output layer to input layer):
15. Determine the gradient of the mistake concerning the weights and biases.
16. Update biases and weights using the gradient and
17. learning rate
18. End For
19. End For

LSTM: Long short term Memory(LSTM) is a specific kind of recurrent neural network architecture designed to address the vanishing gradient problem that ordinary RNNs commonly face. This improvement enables better modeling of long-term dependencies within sequential data. LSTM networks are composed of memory cells equipped with recurrent units that maintain self- connections, facilitating the flow of information across different time steps.

Algorithm: LSTM Algorithm

Inputs

- Training sequences (XTrain, yTrain)
- Learning rate (alpha)
- Number of epochs (numEpochs)
- Number of LSTM units (numUnits)
- Length of input sequences (sequenceLength)
- Number of output classes (numClasses)

Steps

1. Let us initialize LSTM weights and biases the randomly or using specific initialization techniques.
2. Let us Initialize input gate, output gate, and forget gate biases to 1 to facilitatelearning.
3. For epoch = 1 to numEpochs:
4. For each training sequence (XSeq, ySeq) in the training dataset:
5. Let us Initialize LSTM state cell c and (hidden state) h to zeros
6. For t = 1 to sequenceLength:
7. Input gate: Determine the activation function of the input gate
8. Apply sigmoid activation function
9. Forget gate: Determine the activation function of the forget gate
10. Apply sigmoid activation function
11. Output gate: Determine the activation function of the output gate
12. Apply sigmoid activation function
13. Candidate cell state:Determine the activation function of the candidate cell state
14. Apply tanh activation function
15. Cell state update: change the state of the cell using candidate cell state, input gate, forget gate.
16. Hidden state: Determine state using output gate and update cell state
17. Output layer: Compute output of the last LSTM unit
18. Apply softmax activation function to obtain class Probabilities.
19. Backpropagate error through time to update LSTM weights and biases
20. End For
21. End For

RNN: One particular type refer to an artificial neural network designed to process sequential input as a recurrent neural network. RNNs have feed-back loops, which helps them remember information from earlier steps, unlike standard feedforward networks and analyze sequences of inputs. They excel at maintaining a state across different time steps, making them perfect for applications such as time series prediction, language modeling, and speech recognition.

Algorithm: RNN Algorithm

Inputs

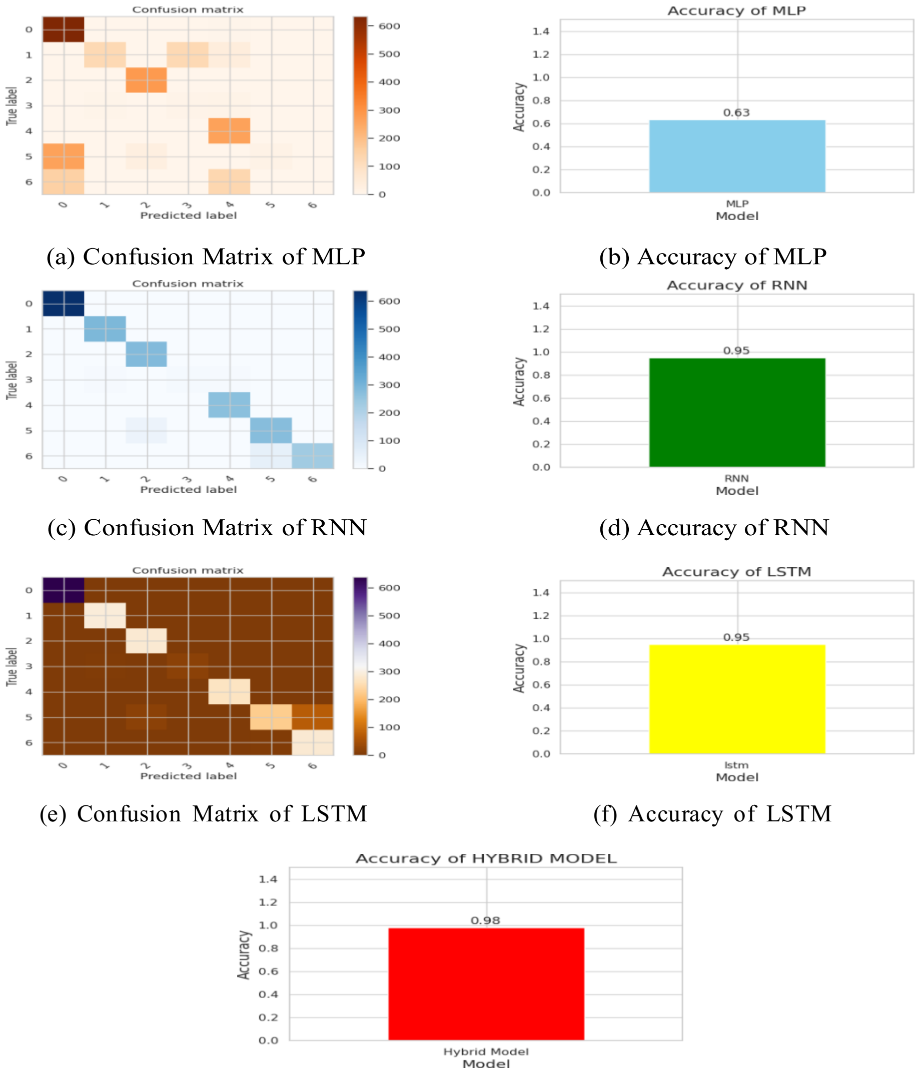
- Training sequences (XTrain, yTrain)
- Learning rate (alpha)
- Number of epochs (numEpochs)
- Number of RNN units (numUnits)
- Length of input sequences (sequenceLength)
- Number of output classes (numClasses)

Steps

```
1. Let us Initialize RNN weights and biases randomly or using specific
   initializa-tion techniques
2. For epoch = 1 to numEpochs:
3. For each training sequence (XSeq, ySeq) in the training dataset:
4. Initialize hidden state h to zeros
5. Forward propagation
6. For t = 1 to sequenceLength:
7. Compute the weighted sum of inputs and hidden States
8. z = W * [XSeq[t], h] + b
9. Apply activation function to compute hidden state:
10. h = activation(z)
11. Output layer
12. Compute output of the last RNN unit
13. Apply softmax activation function to obtain class Probabilities
14. Backpropagation through time (BPTT)
15. Compute error between predicted output and true Output
16. Backpropagate error through time to update RNN weights and biases
17. End For
18. End For
```

4 EXPERIMENTAL RESULTS

This section offers a succinct summary of the experimental setup and explores the outcomes of our proposed framework's classification results. The analysis is supported by metrics including detection accuracies, precision, recall, and F1-score. Additionally, the investigation encompasses training and testing durations, as well as weighted averages obtained from hybrid model classifiers. The hybrid model, composed of individual base classifiers such as MLP, RNN and LSTM, demonstrated exceptional performance across various metrics, accuracies 63,95,95 respectively. And performance metrics and accuracy of the combination of above 3 individual models is 98 are visually shown in Fig.2.



(a) Confusion Matrix of MLP

(b) Accuracy of MLP

(c) Confusion Matrix of RNN

(d) Accuracy of RNN

(e) Confusion Matrix of LSTM

(f) Accuracy of LSTM

(g) Hybrid Model Accuracy

Fig. 2: Confusion Matrices and Accuracies of Models

Table 1. Comparison of Model Accuracies

s.no	Model	Accuracy (%)
1	MLP	63
2	RNN	95
3	LSTM	95
4	Hybrid Model	98

5 CONCLUSION

In this research, we introduce a “Hybrid deep-learning model” that combines several deep neural-network architectures, such as “Multilayer Perceptron”(MLP), “Recurrent neural networks” (RNN), and “Long Short-Term Memory” (LSTM). The algorithm's exceptional performance is demonstrated by the CIC-DDoS2019 dataset findings, which shows 0.64% false alarm rate and 98% accuracy. It is noteworthy that our method's effectiveness is verified on a dataset containing both typical and unusual DDoS attack kinds**.** To make this model even more helpful and reliable in the future,we can detect the location of the attack being initiated and the tool that was used to initiate the DDoS attacks. A thorough quantitative assessment has also beencarried out to evaluate the effectiveness of this model, analyzing its performance metrics in detail.

References

1. H. Aydın, Z. Orman, and M. A. Aydın, “A long short-term memory (LSTM)-based distributed denial of service (DDoS) detection and defense system design in public cloud network environment,”.
2. Y. Wei, J. Jang-Jaccard, F. Sabrina, A. Singh, W. Xu, and S. Camtepe“AE-MLP: A hybrid deep learning approach for DDoS detection and classification,”.
3. Avanija, J., K. E. Kumar, Ch Usha Kumari, G. Naga Jyothi, K. Srujan Raju, and K. Reddy Madhavi. "Enhancing Network Forensic and Deep Learning Mechanism for Internet of Things Networks." (2023). Journal of Scientific & Industrial Research (JSIR), National Institute of Science Communication and Information Resources (NISCAIR) by CSIR, Govt of India, Vol. 82, May 2023, pp. 522-528,2023,DOI: 10.56042/jsir.v82i05.1084
4. I.Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, “Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy,”.
5. I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, “Toward generating a new intrusion detection dataset and intrusion traffic characterization,”.
6. M. S. E. Sayed, N.-A. Le-Khac, M. A. Azer, and A. D. Jurcut, “A flowbased anomaly detection approach with feature selection method against DDoS attacks in SDNs,”.
7. Jyothsna, V., Prasad, K.M., Rajiv, K. et al. Flow based anomaly intrusion detection system using ensemble classifier with Feature Impact Scale.
8. Priyadarshini and R. K. Barik, “A deep learning based intelligent framework to mitigate DDoS attack in fog environment,”.
9. Hassan, K.F.; Manna, M.E. Detection and mitigation of DDoS attacks in the In- ternet of things using a fog computing hybrid approach.
10. Mihoub, A.; Fredj, O.B.; Cheikhrouhou, O.; Derhab, A.; Krichen, M. Denial of service attack detection and mitigation for internet of things using looking-back-enabled machine learning techniques.
11. P. J. Shinde and M. Chatterjee, "A novel approach for classification and detection of DOS attacks".
12. D. Kshirsagar and J. M. Shaikh, "Intrusion detection using rule-based machine learning algorithms".
13. S. Wankhede and D. Kshirsagar, "DoS attack detection using machine learning and neural network".
14. P. Kumari and A. K. Jain, "A comprehensive study of DDoS attacks over IoT network and their countermeasures".
15. Ismail, Muhammad Ismail Mohmand, Hameed Hussain, Ayaz Ali Khan, Ubaid Ullah Muhammad Zakarya, (Senior Member, Ieee), Aftab Ahmed , Mushtaq Raza , Izaz Ur Rahman, And Muhammad Haleem,” A Machine Learning Based Classification and Prediction Technique for DDoS Attacks”.
16. R. SaiSindhuTheja and G. K. Shyam, “An efficient metaheuristic algorithm based feature selection and recurrent neural network for DoS attack detection in cloud computing environment,”.
17. Tin H. Pham And Bijan Raahemi,”Bio-Inspired Feature Selection Algorithms With Their Applications: A Systematic Literature Review”.
18. Jyothsna, V., Munivara Prasad, K., GopiChand, G., Durga Bhavani, D.: DLMHS: Flow-based

intrusion detection system using deep learning neural network and meta-heuristic scale.

19. R. Anusuya,M. Ramkumar Prabhu,Ch. Prathima,J. R. Arun Kumar,"Detection of TCP, UDP and ICMP DDOS attacks in SDN Using Machine Learning approach".