Mass Removal of Botnet Attacks Using Heterogeneous Ensemble Stacking PROSIMA Classifier in IoT

Priyang Bhatt[1*], Bhaskar Thakker[2]

[1*]Gujarat Technological University, Chandkheda, Ahmedabad, Gujarat. India
[2]Symbiosis Institute of Technology (SIT), Symbiosis International Deemed University (SIDU), Pune, Maharashtra, India

Abstract**:** In an Internet of Things (IoT) environment, any object, which is equipped with sensor node and other electronic devices can involve in the communication over wireless network. Hence, this environment is highly vulnerable to Botnet attack. Botnet attack degrades the system performance in a manner difficult to get identified by the IoT network users. The Botnet attack is incredibly difficult to observe and take away in restricted time. there are challenges prevailed in the detection of Botnet attack due to number of reasons such as its unique structurally repetitive nature, performing non uniform and dissimilar activities and invisible nature followed by deleting the record of history. Even though existing mechanisms have taken action against the Botnet attack proactively, it has been observed failing to capture the frequent abnormal activities of Botnet attackers .When number of devices in the IoT environment increases, the existing mechanisms have missed more number of Botnet due to its functional complexity. So this type of attack is very complex in nature and difficult to identify. In order to detect Botnet attack, Heterogeneous Ensemble Stacking PROSIMA classifier is proposed. This takes advantage of cluster sampling in place of conventional random sampling for higher accuracy of prediction. The proposed classifier is tested on an experimental test setup with 20 nodes. The proposed approach enables mass removal of Botnet attack detection with higher accuracy that helps in the IoT environment to maintain the reliability of the entire network.

Keywords**:** Meta-classifier, Ensemble Learning, Botnet, Machine Learning, IoT, PROSIMA-Protein similarity**.**

1. Introduction

A Botnet is applied for cyber-crimes to execute malicious tasks like spam emails, denial-of-service attacks and stealing personal information like mail, accounts, belongings, military secrets, embarrassing info or bank credentials. With the continued rapid advancement of the Internet of Things (IoT), there has been increasing enthusiasm to the understanding of rising digital dangers in IoT domain. IoT devices are amazingly defenseless and alluring to aggressors for their exceptionally heterogeneous parts, innocent security arrangements and powerless encryption check [1]. The term Bot originates from a word Robot that naturally works as per a computer program or contents composed by the Bot master and these Botnet continue to be a significant source of large scale attacks on the Internet with recent increases in the volume of attack traffic [2].

Nodes of the IoT are limited in resources where dedicated, diversified communication protocols are used. Some of these differences weaken the ability of IoT nodes to protect themselves. IoT is connecting smart things, such as intelligent devices and sensors, to the Internet [3]. The data collected by the smart things are sent to a central cloud-based service that processes all the gathered data and shares these data with users [4]. Botnet not only allows the attacker to get access to the device connected with IoT but also get access to the connection. This kind of attack raises security concerns and the control of the IoT device is achieved by a third party for malicious activities. Under such scenarios, lead to the fact that such network devices became another attractive target for cybercriminals [6].

Recently the most powerful attacks were performed by Botnet which consisted mainly on unsecure IoT devices. The Botnet Mirai [7] is considered as the largest Botnet in the history, containing a huge number of compromised IoT devices. C&C servers referred as command and control servers are evolved for providing Botnet management platforms. C & C servers are specialized computers controlled by attackers to send command, spread malicious codes, files and to steal information from victim network [8]. The C&C servers hosting the Botnet herder’s victims are designed to easy deploy a wide array of network and application attacks, provide implementation scripts to Botnet victims, and quickly scale the attacks. The servers are capable of Peer to Peer (P2P) communication and collaboration. The Botnet can then be controlled by single or multiple Botnet herders [9].

The fundamental suspicion of strategies based on machine learning is that Botnet makes discernable patterns inside the system activity and that these patterns could be productively identified utilizing machine learning algorithms [10]. This class of detection approaches guarantees mechanized recognition that can sum up learning about noxious system activity from the accessible perceptions, subsequently dodging traps of mark based discovery approaches that are just ready to identify known movement oddities [11]. For the Botnet attack detection, machine learning algorithms like Random forest, Naive Bayes, SMO and MLP are used for the classification purpose [12].

Support Vector Machine (SVM) is looking for the biggest factual edge in the interim that keeps near each other from a similar class and far away to each other from the diverse classes in the edge sense. [13]. Fuzzy means clustering algorithm likewise is also utilized for characterizing information. SVM, Random Forest, and Naïve Bayes with customary word vectors, an LDA-based classifier has better execution. The downstream of machine learning examination is an expansion for the learning approach yet to be considered [14]. Larger number of IoT devices connected with the Internet creates an issue of security and makes the situation vulnerable to the Botnet attack [15]. Approach is well suited for detecting compromised IoT devices, because these connected appliances are typically task-oriented. Accordingly, they execute fewer, and potentially less, complex network protocols, and exhibit traffic with less variance than PCs, however, the prediction accuracy is very low [16].

IoT could be a distinct network with sizable amount of applications wherever there's an opportunity in prevalence of traffic and privacy considerations whereas single degradation of a system fails out the entire structure. Similarly, hackers intrude the system using Botnet and degrade the system. Therefore it becomes essential to observe the Botnet accurately and to frame out a structure to induce obviate Botnet. In the IoT working with high dimensional data will cause delay in the Botnet detection. Delay in the Botnet detection will slow down entire the performance of entire network.

Section 2 of the paper focuses on related research of on Botnet attacks. Section 3 describes proposed methodology and section 4 displays implementation of the proposed methodology and result analysis. Sections 5 deals with conclusions arrived.

2. Related Research

Meidan, Yair, Michael Bohadana, Yael Mathov et al. proposed auto encoders for anomaly detection from network traffic. Botnet attacks have been detected from compromised IoT devices with high accuracy and very false error rate. Auto encoder built for each and every IoT device in the network was trained with malicious network traffic [17]. McDermott, D.Christopher et al. proposed deep learning based Bidirectional Long Short Term Memory based Recurrent Neural Network (BLSTM-RNN) model to detect Botnet for IoT devices. BLSTM-RNN was used to recognize the text and attack vector was converted into tokenized integer format. That was how less FPR (False positive rate) in Botnet attack detection [18].

YairMeidan , Michael Bohadana and AsafShabtai presented machine learning algorithms on network traffic data for accurate identification of IoT devices connected to a network. To train and assess the classifier, it collected and labeled network traffic data from nine distinct IoT devices, and PCs and smartphones. Consuming supervised learning, it trained a multi-stage Meta classifier; in the first stage, the classifier can distinguish between traffic generated by IoT and non-IoT devices. In the second stage, each IoT device was linked a specific IoT device class [19].

Homayoun, Sajad, Marzieh Ahmadzadeh, et al. proposed BoTShark Deep learning based Botnet traffic shark using Convolution Neural Network (CNN) and used Softmax at the end to identify malicious traffic. That was how attacks from compromised IoT devices were detected [20]. An, N., Duff, A., Naik, G., Faloutsos, et al. collected data from darknet and applied multiple supervised machine learning algorithms to identify malicious IoT devices from IoT Network [21].

Lakshya Mathur et al. performed different experiments on different classifier techniques to detect Botnet attacks. Randomized filtered classifiers, logistic regression, random committee, Random subspace machine learning algorithms were implemented [22]. Francisco Villegas Alejandre et al. defined the feature selection process for efficient detection of the Botnet attack in the network. The main aim of this paper was to support different researchers to select different efficient features from the dataset to improve accuracy to detect Botnet attack [23]. Anchit Bijalwan et al. performed a Botnet analysis using an ensemble classifier on ISCX dataset. The entire dataset was divided into Botnet traffic or not after performing the future extraction. Experiment on KNN, Decision Tree, bagging with KNN, Adaboost classifiers were performed [24]. Sean Miller and Curtis Busby-Earle provided the survey of different machine learning techniques which can be used for detecting Botnet attack [26]. Hammerschmidt, C., Marchal, S. al. Proposed solution created a solution using finite state machines and network flow features to detect devices infected by bot malwares [27]. Kirubavathi Venkatesh and Anitha Nadarajan [28] has detected the Spyeye and Zeus Botnet with the aid of adaptive learning rate multilayer feed-forward neural network. Here in this work, various classifiers such as Decision tree, Random forest and radial basis function are discussed and are compared with the actively learned neural network.

Kamaldeep Singh et al. [29] built a random forest based decision tree model, to solve the problem of Botnet detection in a peer-to-peer network. Though the method served good for detecting Botnet, it has failed to detect Botnet under low frequency communication, during when certain threshold exceeded.

The system enabled effective detection of malicious activities among nodes but they could not differentiate the kind of attack performed by the Botnet also they didn't get data from massive traffic information set, while our work concerns to cluster out similar kind of Botnet attacks with help of PROSIMA and with higher accuracy because utilization of stacking classifies [17].The proposed system had high Botnet detection rate. But because of deep learning approach time complexity was very high. Our proposed work cluster out the similar kind of Botnet with high detection rate with less time complexity [18].The proposed system had high classification ratio. But due to multistage time complexity was high. [19]. the proposed system could work on encrypted data. But it failed during large dataset [20].The system had higher accuracy. But the same time, time complexity was also high [21]. The system was proposed for Botnet detection with SVM. But on the specific kind of Botnet attack like MrBlack and Mirai [22]. This paper was to support different researchers to select different efficient features from the dataset to improve accuracy to detect Botnet attack [23]. Experiment was performed on ISCX data using Ensemble classifiers using KNN and Decision Tree. Main motivation behind the analysis was missing [24]. Survey of different machine learning techniques which can be used for detecting Botnet attack but real Botnet detection was missing [25]. Different flow feature selection for supervised as well as unsupervised learning algorithms for Botnet detection, that can be used my other researcher to detect the Botnet attack [26]. The authors state that recent methods that deal with Botnet detection work in a batch setting, which creates time and memory constraints. In this sense, they propose to adequate their approach to deal with network data in a stream setting. When evaluating their approach they achieved high host identification rates; however, their solution does not identify malicious Botnet flows, requiring a high number of flows to perform the detection task [27]. On the basis of related research surveyed, assumed that further improvement is required in the detection process of Botnet in the IoT based network. So learning based classifier is proposed to detect and cluster the similar sources to help in mass removal of Botnet attacks. So it maintains the reliability of entire IoT network.

In conclusion, major contribution of proposed work contains three stages,

- First collecting the attacker activity pattern from the IoT devices serve as honeypot, but in the conventional Botnet detection techniques relies on network traffic flow as well as behavior analysis.
- Second stage of work, modeling the attacker information in tree-based structure by stacking classifier.
- In the third stage similar kind of attacks would be clustered by PROSIMA protein similarity.

3. Heterogeneous Ensemble stacking PROSIMA classifier

In IoT environment, Botnet attack is carried out by nodes which are compromised, so it is very difficult to detect Botnet compromised nodes. In our proposed model data is collected from the different sensor nodes and unwanted data are removed during preprocessing stage. The preprocessed data are used for training in heterogeneous Ensemble stacking classifier. In the phase two of the proposed classifier, again random forest algorithm is used as a meta-classifier. In the testing phase, similar Botnet would be clustered by PROSIMA protein sequence similarity algorithm. Figure 1 shows the proposed heterogeneous ensemble stacking PROSIMA classifier. Figure 2 shows the overall structure of network scenario under consideration. In this proposed approach, the classifier is used at gateway but it can be used in the node as well, if the node is capable enough to carry pre-trained model of the classifier.

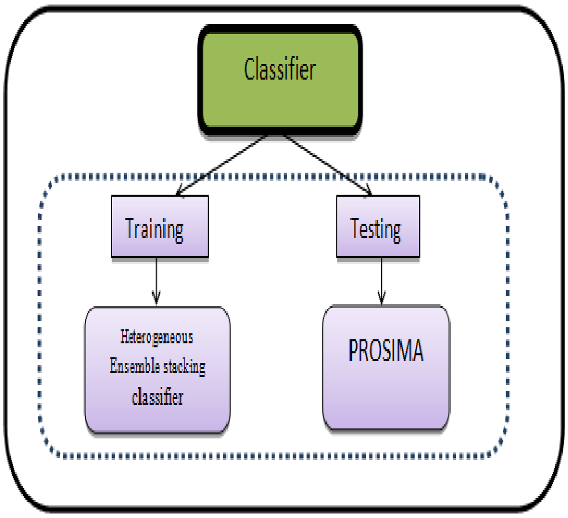


Figure 1. Heterogeneous Ensemble stacking PROSIMA classifier

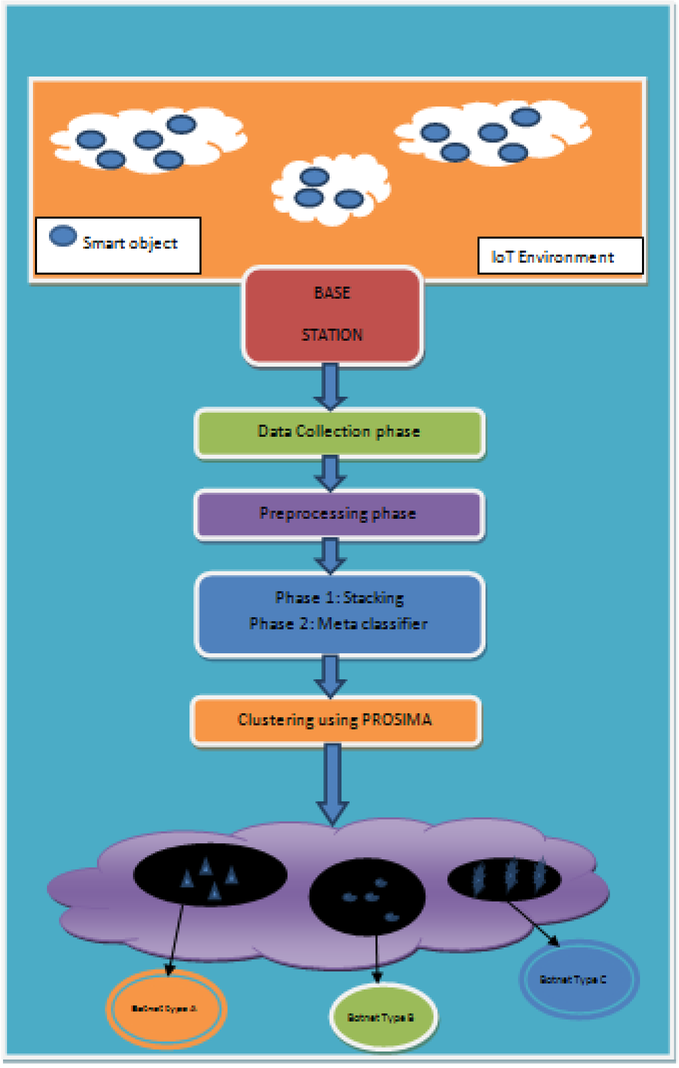


Figure 2. Overall Architecture of the IoT network with Botnet attack

3.1 Data Collection

In the experimental setup, each IoT node is connected with a sensor node Sn. Data are generated at every sensor nodes Sn= (S0, S1…………Sn).The collection of resources are identified as R= {IoT1, IoT2, IoT3… IoTn}. Since data are collected from the sensor nodes, it includes raw data along with network traffic. In order to remove the unwanted as well as redundant information from the collected data, data preprocessing is required.

3.2 Preprocessing

The data packets arrived are captured with the help of Wireshark in the form of 'pcap' file. With the help of Tshark command, the 'pcap' file is converted to CSV file. The features required for analyzing the packets for the response for the presence of Botnet attack are derived from CSV file. Total 21 features like arrival time, source, destination, protocol, length etc. are derived for this purpose. In order to detect Botnet, only network traffic information would be required. So it is necessary to remove unwanted information like sensor data. Sensors data are removed, and only network traffic flow is retained in the feature set. XGBoost, Adaboost and Random forest were used to avoid value scaling.

3.3 Feature selection

Feature Selection is the method of finding the most relevant features from available feature set for a classifier model. These techniques are accustomed to establishing and take away needless, tangential and redundant options that don't contribute or decrease the accuracy of the model. Most powerful technique would be a Genetic algorithm. After preprocessing stage, genetic algorithm is used for identifying relevant features selection to visualize the data as well as to reduce processing time further for classification stage. The first step is to form and initialize the individual within the population. Because the genetic algorithmic program may be a random improvement technique, the genes of the people area unit are sometimes initialized haphazardly. In the second stage would be assigned fitness value to each individual. The model is trained with entire training dataset to evaluate the fitness. Fitness values would be assigned by Rank based method.

The fitness value is assigned to individuals using Rank based method as following:

$$\emptyset(i) = k * R(i) \quad i = 1, \ldots, n \qquad (1)$$

Here k is constant and also called selective pressure. Its value is fixed between 1 and 2. In the proposed work this value is selected to 1 as per literature of Genetic Algorithm. Greater selective pressure values can create the fittest individual to own a lot of chance of recombination. The parameter R (i) is the rank of individual ‘i’.

$$R(i) = \frac{rank(i)}{n*(n-1)} \qquad (2)$$

Once the fitness assignment is performed, the choice operator chooses the individual that may recombine for the following generation. Therefore, the selection operator selects the individual in step with fitness level for the next crossover. Next, the GA can determine how bits are swapped among the try. After receiving fitness value, feature selection is performed using Mod-Dejong on our dataset. Mod-Dejong gives 4 features (arrival time, packet delivery ratio, packet loss and throughput) which would be utilized to training the proposed algorithm.

3.4 Proposed Model

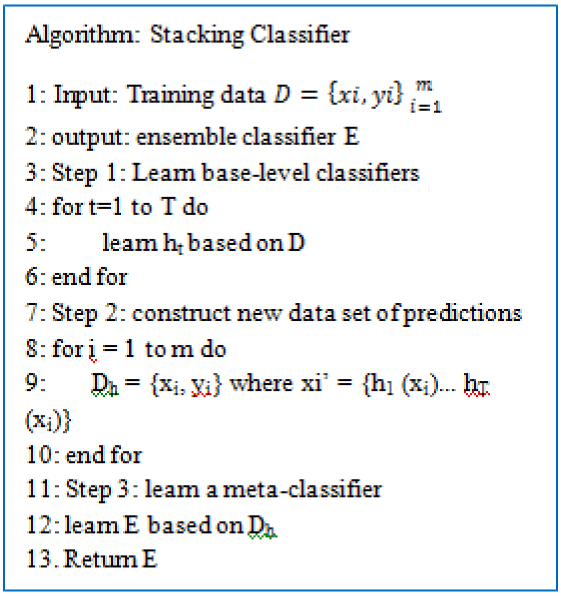
3.4.1 Popular ways to combine different classifiers

There are classifiers which are showing results to identify present of Botnet attack with different methodology. Popular approaches in which of different classifiers can be combined are by bagging, boosting and voting. This is also referred to as ensemble learning. Bagging, Boosting and Voting would be the popular way of combining totally different classifiers and trained them on random subset of the data called ensemble learning [6]. One of the examples of bagging is random forest. Boosting which is very similar to the bagging but here in bagging previous bag errors are taken into consideration. One of the examples of boosting is Adaboost. Bagging is better than boosting. Boosting can lead to over fitting in the classifier. Where model works better on the training data set but fails to detect the attack on unknown data. There are two main techniques to combine the model, voting and stacking. In voting, the class is predicted as a majority vote from the different classifier. Stacking classifier is discussed in the next section.

3.4.2 Stacking classifiers

The main advantages of using stacking classifier is the outputs of the base level classifiers area unit then accustomed train a Meta classifier. The goal of this next level is to confirm the learning process. For example, if a classifier consistently misclassifies the instances from one region as a result of incorrect learning of the feature area of that region, the Meta classifier could be ready to discover this downside. Exploiting the learned behaviors of alternative classifiers, it will improve such learning deficiencies.

Stacking is the process of combining different classifiers CL1, CL2 ..., CLn on the single dataset. It is a two steps process. In the first step, a set of base classifiers BC1, BC2…, BCn is used. In the second step, a Meta classifier is used which performs predictions on newly constructed dataset.



```
Algorithm: Stacking Classifier

Input: Training data $D = \{xi, yi\}_{i=1}^{m}$
output: ensemble classifier E
Step 1: Learn base-level classifiers
for t=1 to T do
        learn $h_t$ based on D
end for
Step 2: construct new data set of predictions
for i = 1 to m do
        $D_h = \{x_i, y_i\}$ where xi' = $\{h_1(x_i) \ldots h_T(x_i)\}$
end for
Step 3: learn a meta-classifier
learn E based on $D_h$
Return E
```

3.4.3 Overall Architecture of Heterogeneous Ensemble Stacking Meta-classifier

In proposed Heterogeneous Ensemble Stacking Meta-classifier, XGboost, AdaBoost and Random forest heterogeneous classifiers are used. Again Random forest classifier is used as Meta level classifier. During testing phase similar Botnet are clustered using PROSIMA (Protein Similarity) algorithm.

3.4.3.1 XGBoost (Extreme Gradient Boosting) Algorithm

XGBoost is an associate algorithmic program that has recently been identified as dominating for applied machine learning and Kaggle competitions for generating structured or tabular information. XGBoost is an associate implementation of gradient boosted call trees designed for achieving higher amount of speed and performance simultaneously. The sweetness of this powerful algorithmic program lies in its measurability that drives quick learning through parallel and distributed computing and offers economical memory usage.

3.4.3.2 Adaboost (Adaptive Boosting) Algorithm

Adaboost is a preferred algorithm to boost the performance of call trees on binary classification issues. It is stated as distinct Adaboost as a result of its use for classification instead of regression. It is best used with weak learners.

3.4.3.3 Random cluster sampling forest Algorithm

Random forest builds multiple decision trees and merges them along to induce an additional correct and stable prediction. Here is given the algorithmic rule for random forest algorithm. Due to its performance and accuracy, Random forest is used both as base classifier and Meta classifier.

The conventional random forest takes less time to train but more time for predictions because large number of trees would slow down the performance of the algorithm. So cluster sampling is adopted in place of random sampling in the Meta classifier stage to speed up the prediction process. Figure 3 shows the training process of random clustering forest.

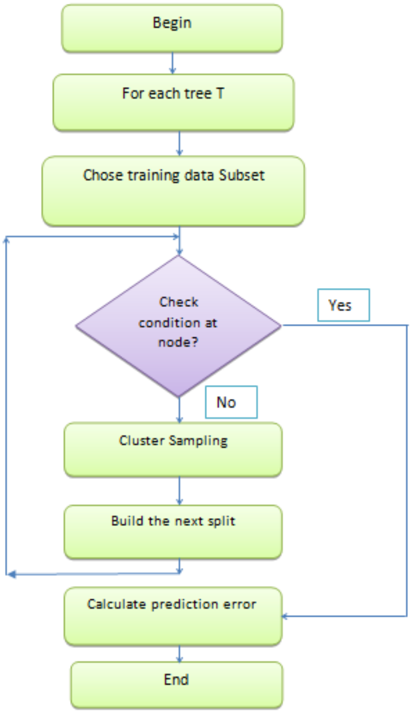


Figure 3. Training process with random clustering sampling forest algorithm

Prediction of unseen sample using Random Forest is defines as:

$$F' = \frac{1}{T}\sum_{t-1}^{T} F_t(E(s)) \qquad (3)$$

F' indicates the prediction of all the unseen samples and Ft indicates the time period for observation; E(s) represents the Poisson distribution of trained data set which reduces the time of training. Bagging process repeatedly (T times) selected the random sample from the training dataset.

The main significance of this (Random forest) model is that instead of searching down the simplest feature whereas half a hub, it scans for the simplest feature among Associate in nursing irregular set of features. This procedure makes it the best model. Figure 4 shows process flow of Heterogeneous Ensemble staking meta-classifier. Figure 5 shows flow of proposed system. Figure 6 shows clustering of Botnet using PROSIMA.

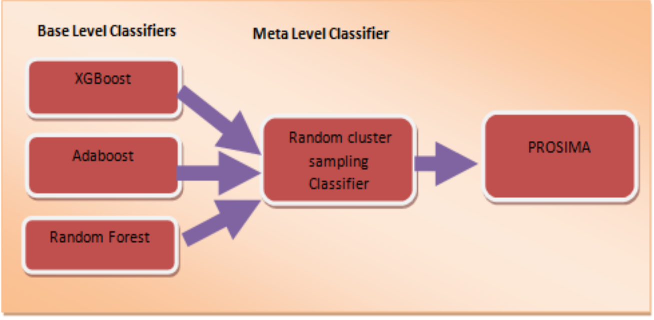


Figure 4. Process flow of Heterogeneous Ensemble stacking meta-classifier

3.5 Mass clustering based on PROSIMA protein similarity

All the similar Botnet having repetitive structure is clustered by PROSIMA protein similarity algorithm. Output of the training phase eq (2) is clustered in the testing phase.

We use m different terms t1, t2…..tm for indexing N features. Then each observation Oi is represented by a Vector:

$$di = \{Oi1, Oi2, Oi3 \ldots\ldots. Oin\} \qquad (4)$$

Where, Oij is the weight of the term tj in the observation di. An index file of the vector model is represented by matrix:

$$D = \begin{bmatrix} O11 & O12 & \ldots & O1m \\ O21 & O22 & \ldots & O2m \\ \vdots & \vdots & \ddots & \vdots \\ On1 & On2 & \ldots & ONm \end{bmatrix} \qquad (5)$$

Where, ith row matches ith observation and jth columns matches jth term. The Similarity of two observations is given by following formula,

$$sim(di, dj) = \frac{\sum_{k=1}^{m}(Oik\ Ojk)}{\sqrt{\sum_{k=1}^{m}(Oik)^2}\sqrt{\sum_{k=1}^{m}(Ojk)^2}} \qquad (6)$$

1. Input generalized suffix tree data structure from Meta Level classifier
2. Find all maximal substructure clusters within the suffix tree.
3. Build a vector model of all pockets in our assortment
4. Build pocket similarity matrix

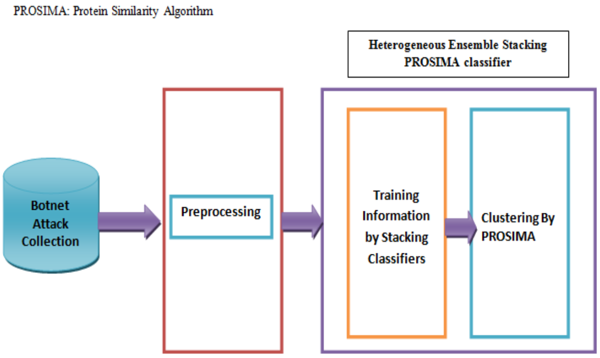

PROSIMA: Protein Similarity Algorithm
Heterogeneous Ensemble Stacking PROSIMA classifier
Botnet Attack Collection
Preprocessing
Training Information by Stacking Classifiers
Clustering By PROSIMA

Figure 5. Process flow of the proposed system

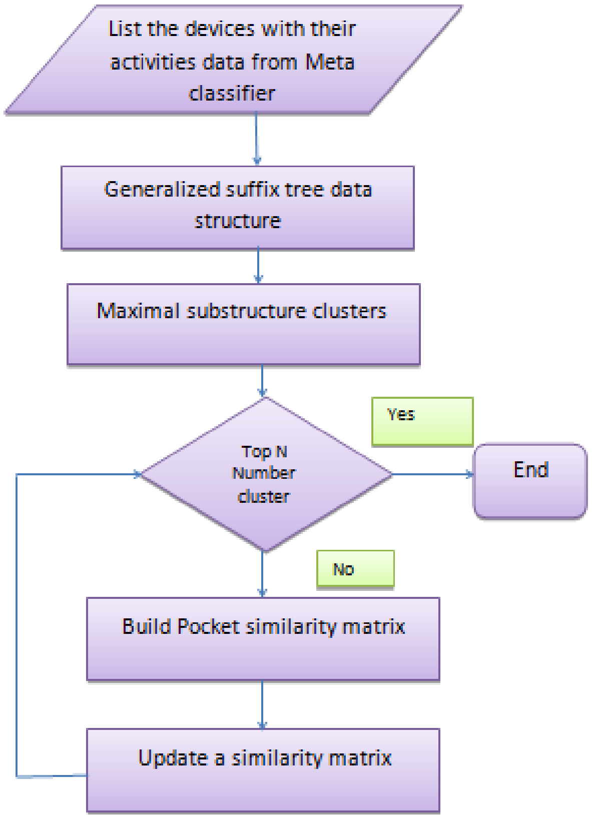

List the devices with their activities data from Meta classifier
Generalized suffix tree data structure
Maximal substructure clusters
Top N Number cluster
Yes
End
No
Build Pocket similarity matrix
Update a similarity matrix

Figure 6. Flow diagram for clustering of Botnet using PROSIMA

4. Result Analysis

In the experimentation, two kinds of attack are considered of. They are Distributed denial of service attack (DDoS attack) and spam attack, DDoS attack may be a digital attack during which the attacker tries to make a machine or system inaccessible by incidentally or inconclusively distressful administrations of a bunch related to the Internet.

Email Spam is an electronic type of garbage mail. It includes undesirable messages, often spontaneous Business enterprise. Spam may be a real security worry because it will be used to convey Trojan stallions, Infections, worms, spyware, and targeted on phishing attacks. In normal attack single attacker would try to disturb the network. But in the case of Botnet attack number of malicious nodes called as Bot (Compromised node), would be trying to attack the target system as a whole with every connected node getting affected.

The proposed method is evaluated with the experimental setup. The traffic is collected from 20 IoT nodes (implemented with Raspberry pi 3) connected via WI-FI network to the access point and wired connection to the central switch and also to the router. Using Tshark and Wireshark the network traffic is sniffed, port mirroring on the switch has been utilized for sniffing. C & C (command & control) has been achieved using python script to send the file and to control the IoT devices. Three IoT devices are configured a bots to generate DDoS and Spam attacks to the rest of the devices in the network. Twenty one features have been extracted from 5 time windows each of 50ms, 100ms, 500ms, 10ms and 1.5ms respectively. Using python script and Tshark commands, packet delivery ratio, packet loss, and throughput, packet arrival time as number of received/sent packets are computed. Arrival time is computed as shown 4.1.1. Figure 7: Shows the experimental setup for detecting Botnet attacks.

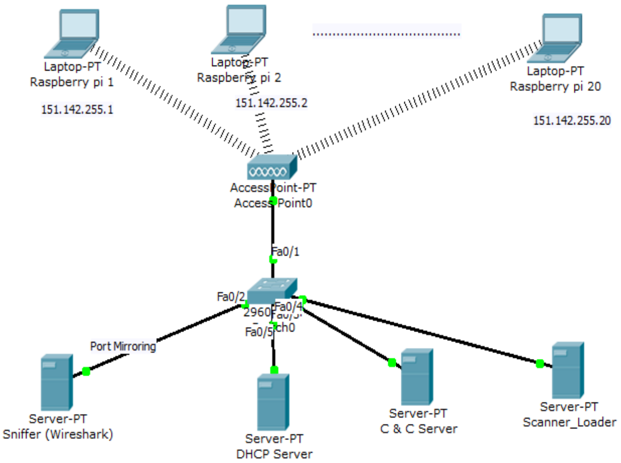
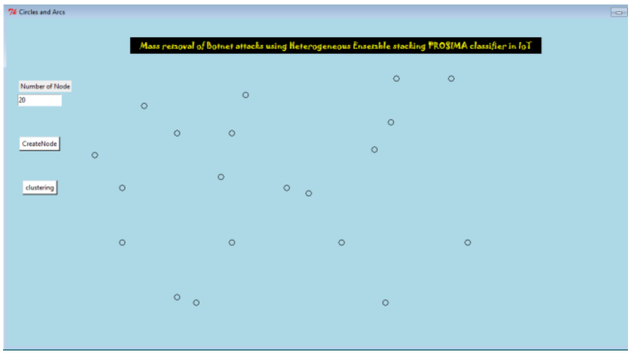


Figure 7. The experimental setup for detecting Botnet attacks

To send the DDoS and Spam attacker script on the IoT devices (Raspberry pi 3), brute forcing is carried out Telnet port. In the proposed work, IoT devices are infected using created DDoS and Spam attacks. Required python scripts are created using python Scapy. Under the influence of attack the IoT devices started generating DDoS and Spam attacks for the rest of the devices available in the network. The result of one of such experiment is shown in table 2. The traffic data collected for the experimental setup has been further utilized to carry out performance evaluation of the proposed classifier.

4.1 Implementation

The Proposed system for IoT based network is implemented using python programming language. Figure 8 shows the IoT based network implemented using python.


Mass removal of Botnet attacks using Heterogeneous Ensemble stacking PROSIMA classifier in IoT

Number of Node: 20

Figure 8. IoT based network

4.1.1 Calculation of Packet arrival time

Since smart objects are involved in the IoT based network, In absence of security unauthorised users can easily get the access of the network and IoT node resources and distribute the false information which affects the working of the IoT node. In Botnet attack, smart objects carry out the malicious activities by forming group among each other, so by keeping track of arrival and inter arrival time of each smart object which is involved in the IoT based network, suspicious users can be listed and monitoring process would be executed on those users.

Let $Q(t)$ be a process with rate (λ) .Let V_1 be the time of the first arrival, then

$$P(V_1 > t) = P(no\ arrival\ in(o,t))$$

$$P(V_1 > t) = e^{-\lambda t}$$

$$F_{v1}(t) = \{1 - e^{-\lambda t}; t > 0\} \qquad (7)$$

So V_1~ Exponential(λ), Let V_2 be the time interval between the first and second arrival

And let $S > 0$ and $t > 0$, two intervals $(0,S)$ $(S,S+t)$ are independe

$$P(V_2 > t)/V_1 = S) = P(no\ \ arrivals\ in\ \ (S,S+t)/V_1 = S)$$

$$= P(no\ arrivals\ in\ \ (S,S+t) = e^{-\lambda t} \qquad (8)$$

If $Q(t)$ be a process with rate (λ), then the inter arrival times $V_1 V_2$ are independent

V_1~ Exponential (λ), for i=1, 2…

R_n is the sum of n independent exponential (λ) random variables then:

$$R_n = V_1 + V_2 + \ldots\ldots + V_n$$

The Probability Density Function of R_n for n=1, 2, 3…

$$f\,R_n(t) = \frac{\lambda^n R^{n-1} e^{-\lambda t}}{(n-1)!} \qquad (9)$$

If X ~ Exponential (λ),then

$$E(X) = \frac{1}{\lambda}$$

$$Var(X) = \frac{1}{\lambda^2}$$

Since $R_n = V_1 + V_2 + \ldots\ldots\ldots\ldots V_n$, it is concluded that arrival time of the Poisson distribution is calculated by

$$E(R_n) = nEV_1 = \frac{n}{\lambda} \qquad (10)$$

$$Var(R_n) = nVar(V_n) = \frac{n}{\lambda^2} \qquad (11)$$

In IoT based network, arrival time of each smart object would be calculated based on the arrival time.

$E(R_n)$ Indicates the mean arrival time of each user in the IoT based network.

4.1.2 Packet Delivery Ratio

The estimation of Packet Delivery Ratio (PDR) depends on the received and created bundles as recorded in the trace document. All in all, PDR is characterized as the proportion between the got bundles by the goal and the created parcels by the source.Figure 9 shows packet delivery ratio during normal and attack period.

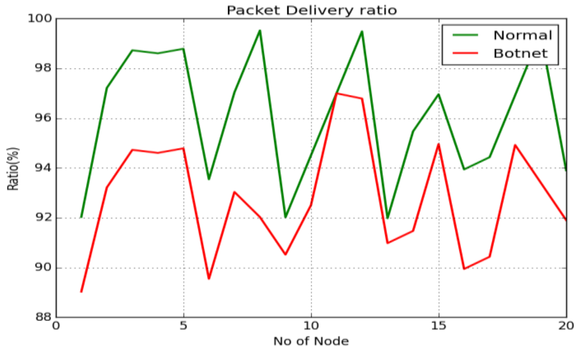


Figure 9. Packet delivery ratio during normal and attack period

4.1.3 Packet Loss

Packet loss happens once at least of one packet of knowledge traversing a network fails to attain its goal. Packet loss is calculated as tier of packets lost with reference to packets sent. Figure 10 shows the packet loss ratio of the network during normal flow and attack.

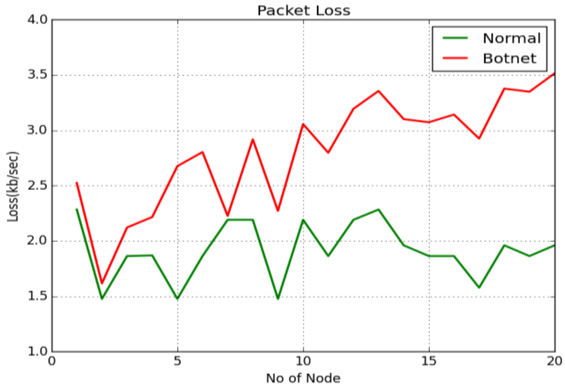


Figure 10. Packet loss ratio of the network during normal flow and attack

4.1.4 Throughput

In data transmission, throughput is the quantity of information transferred with success from supply node to destination node in an exceedingly nominal period, and usually measured in bits per second (bps), as in megabits per second (Mbps) or gigabits per second (Gbps). Figure 11 shows that throughput of the network under normal and attacked period.

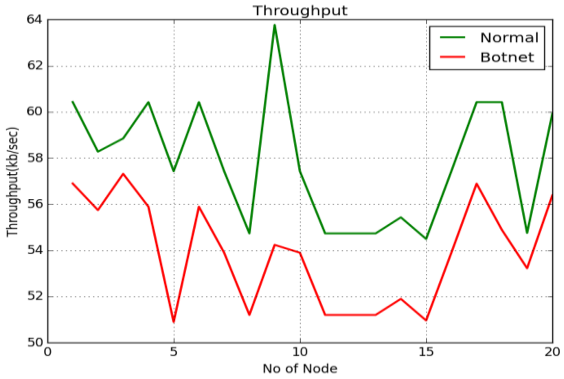


Figure 11. Throughput ratio of the network during normal flow and attack

Table 1. Log details of each smart node in the network

Node	IP Address	Arrival Time (Sec.)	Packet Delivery Ratio	Packet Loss
n1	151.142.255.1	2.256	88.025	2.2835
n2	151.142.255.2	1.267	93.211	1.4756
n3	151.142.255.3	8.278	94.723	1.8629
n4	151.142.255.4	1.289	94.601	1.8687
n5	151.142.255.5	1.314	94.783	1.4756
n6	151.142.255.6	5.311	89.5404	1.8629
n7	151.142.255.7	4.322	93.031	2.1905
n8	151.142.255.8	3.333	95.5216	2.1905
n9	151.142.255.9	2.344	88.0122	1.4756
n10	151.142.255.10	1.355	90.5028	2.1905
n11	151.142.255.11	1.366	92.9934	1.8629
n12	151.142.255.12	9.377	95.484	2.1905
n13	151.142.255.13	8.388	87.9746	2.2835
n14	151.142.255.14	7.399	91.4652	1.9597
n15	151.142.255.15	6.441	92.9558	1.8629
n16	151.142.255.16	5.421	89.937	1.8629
n17	151.142.255.17	4.432	90.4276	1.5771
n18	151.142.255.18	3.443	92.9182	1.9598
n19	151.142.255.19	2.454	95.4088	1.8629
n20	151.142.255.20	1.465	89.8994	1.9598

4.2 Clustering of Botnet of DDoS and Spam attack

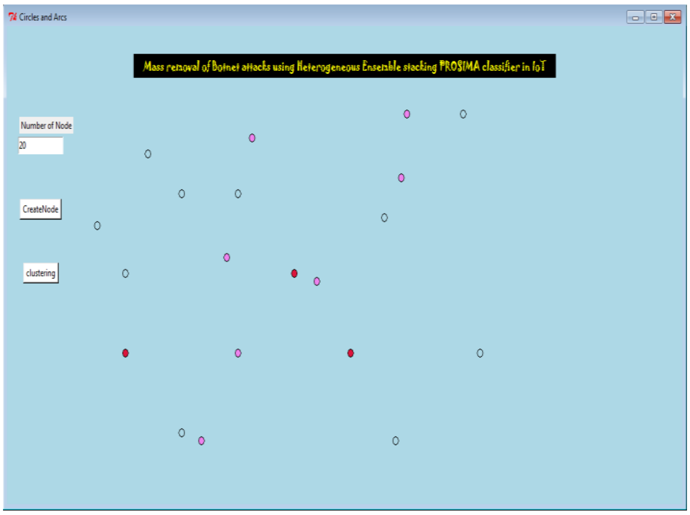
In DDoS types of Botnet attack, cluster of attackers would send the request for resource to constant destination address for such time unendingly therefore authenticate user cannot get that resource for a specific time. The proposed classifier would cluster those nodes supporting the similarity worth of packet sending time, destination address and also the resource that they requested unendingly and also the distance between source nodes and destination node is calculated so as to expeditiously cluster the attacks.

Spam types of Botnet would send the e-mail to the spam box rather than causing to the inbox of the mail application. It includes causing undesirable messages. Spam could be a real security worry because it may be used to convey Trojan stallions, infections, worms, spyware, and centered on phishing attacks.

In the existing techniques hierarchical cluster and K-means clustering have been used. Main drawback of the hierarchical cluster is that if two clusters join together, that cannot be disjoined, and K-means would be required to know K values prior to the execution of the algorithm. In the proposed system mixture model clustering has been used which is similarity-based clustering with the higher clustering ratio compared to existing system. Mixture model clustering would be able to handle any kind of cluster shapes. Figure 12 shows the clustering of Botnet attacks which leads to Distributed DoS attack and SPAM attack.

Table 2. Shows that list of nodes clustered under DDoS Botnet and SPAM Botnet attack

Node	Source IP Address	Packet Sending Time(sec)	destination IP Address	Resource
n1	151.142.255.1	0.214	151.142.250.11	file-1
n2	151.142.255.2	0.214	151.142.250.11	file-1
n3	151.142.255.3	0.214	151.142.250.11	file-1
n4	151.142.255.4	0.214	151.142.250.11	file-1
n5	151.142.255.5	0.214	151.142.250.11	file-1
n6	151.142.255.6	0.214	151.142.250.11	file-1
n7	151.142.255.7	0.214	151.142.250.11	file-1
n8	151.142.255.11	0.114	151.142.255.1	mail
n9	151.142.255.12	0.114	151.142.255.1	mail
n10	151.142.255.13	0.114	151.142.255.1	mail
n11	151.142.255.1	0.214	151.142.250.11	file-1
n12	151.142.255.2	0.214	151.142.250.11	file-1
n13	151.142.255.3	0.214	151.142.250.11	file-1



Circles and Arcs

Mass removal of Botnet attacks using Heterogeneous Ensemble stacking PROSIMA classifier in IoT

Number of Node: 20

Figure 12. Clustering of Botnet attack which leads to Distributed DoS attack and SPAM attack

As shown in the figure 12 seven nodes are clustered under DDoS attack (Pink colored) and three nodes are clustered under SPAM attack (Red Colored).

4.3 Comparing proposed classifier with existing classifiers

In this section, the proposed classifier is compared with existing classifiers in terms of different parameters like Precision, Recall, F-measure and Accuracy.

Table 3. Shows list of classifiers with proposed system [28]

Classifiers	Precision	Recall	F-Measure	Accuracy
IoTDS [30]	0.968	0.931	0.949	96.5333
BoTshark [20]	0.968	0.934	0.95	96.667
Proposed	0.971	0.963	0.966	98.63

Classifiers	Precision	Recall	F-Measure	Accuracy
Decision Tree	0.968	0.931	0.949	96.53
Random Forest [29]	0.968	0.934	0.95	96.66
RBF	0.976	0.927	0.95	96.53
Proposed	0.971	0.963	0.966	98.63

Precision

Precision is revels what fraction of test part of the data is detected as attack is literally from the attack categories. Figure 13 shows comparison graph for precision for four types of classifiers.

$$Precision = \frac{True\ Poistive}{True\ Poistive + False\ Positive}$$

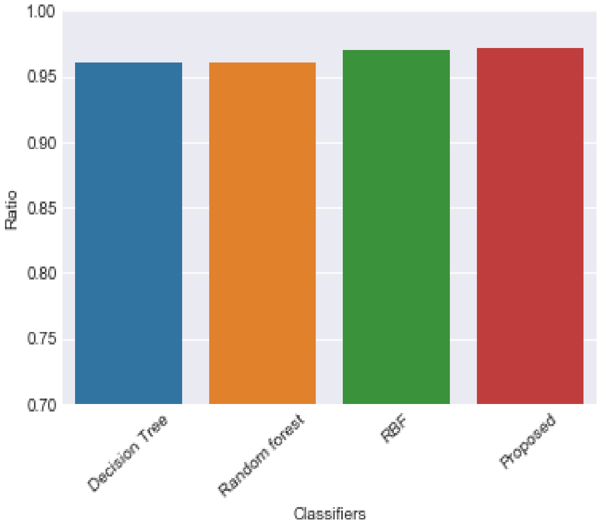


Figure 13. Comparison graph for Precision

The proposed classifier achieved an optimum precision value of 0.971. Comparatively precision value is better than existing classifiers. Since Meta-classifier has adapted a cluster-based sampling approach, which first finds similar elements and then splitting is performed.

Recall

Recall measures the fraction of attack class that was correctly detected. Figure 14 shows comparison graph for recall.

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

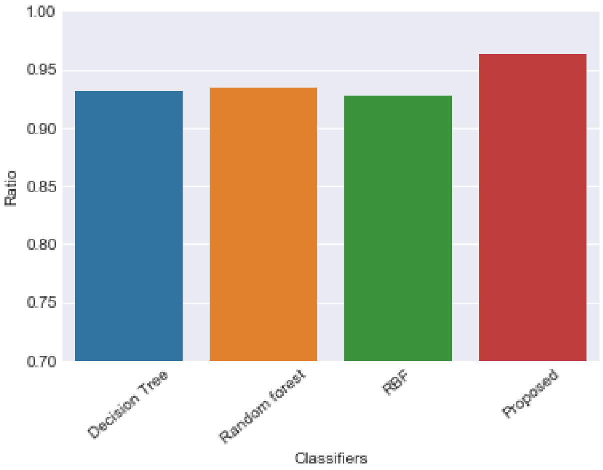


Figure 14. Comparison graph for Recall

The Proposed classifier achieved better recall value of 0.963 compared to other existing classifiers decision tree, random forest, RBF having precision values 0.931, 0.934 and 0.927 respectively. The proposed system has utilized similarity-based clustering. So, it separates the event successfully.

F-Measure

F-measure is a measure of a measure of test's accuracy, which measures the balance between precision and recall. Figure 15 shows comparison graph for F-measure.

$$F\ Measure = \frac{2 * (precision * recall)}{precision * recall}$$

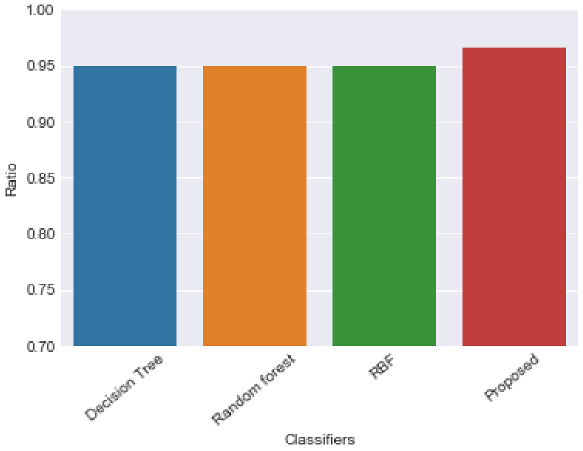


Figure 15. Comparison graph for Recall

The proposed system has utilized cluster-based sampling in the training phase. It first clusters out a similar event before performing splitting the observation for decision tree creation. So, it achieved better F-Measure compare to existing classifiers.

Accuracy

Accuracy is that the portion of predictions our model got right. Formally, accuracy can be defined as,

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ Number\ of\ predicions}$$

For Binary classification problem accuracy is defined as,

$$Accuracy = \frac{True\ Positive + True\ Negative}{True\ Positive + True\ Negative + False\ Positive + False\ Negative}$$

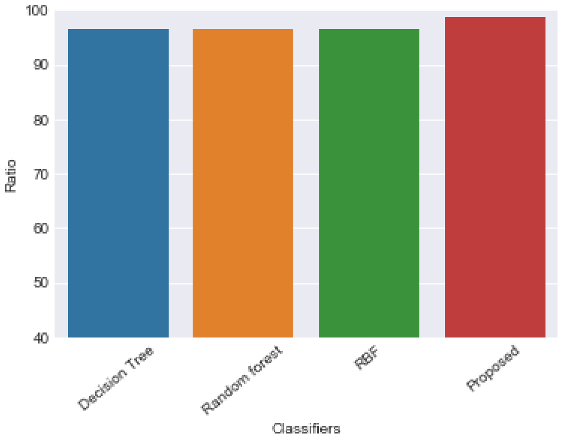


Figure 16. Comparison graph for Accuracy

The proposed classifier has utilized top class base classifiers at the first phase and Meta classifier with cluster-based sampling at the second stage. Then similar Botnet would be clustered by PROSIMA based on similar pocket value.

So proposed classifier has qualified higher accuracy of 98.63 compared to existing classifiers Decision Tree, Random forest and RBF had 96.53, 96.66 and 96.53.

5. Conclusions

IoT based applications such as smart home, smart city, connected health, smart supply chain and smart farming are based on the context-aware computing where an application would sense the physical environment and change their function accordingly, providing comprehensive information security is a challenging one and an integral part of the whole IoT based system. In this work, heterogeneous ensemble stacking classifier has been used, so prediction rate is high. Clustering the same kind of Botnet from the trained data set using PROSIMA classifier enables bulk removal of Botnet. The proposed system achieves more reliability of the IoT based network by removing Distributed Denial of Service (DDoS) and spam Botnet. As a future work, we plan to use another approach as well as more IoT botnet attack datasets to analyze the proposed approach as well as conduct comprehensive comparisons for IoT botnet attack detection.

References

[1] Q. Yan, W. Huang, X. Luo, F. Richard Yu, “A multi-level DDoS mitigation framework for the industrial Internet of things,” IEEE Communications Magazine, Vol.56, No.2, pp.30-36, 2018.

[2] M. Yeo, Y. Koo, Y. Yoon, T. Hwang, J. Ryu, J. Song, C. Park, ”Flow-based malware detection using convolution neural network,” IEEE Information Networking (ICOIN), pp. 910-913, 2018

[3] S. Wook Park, J. Park, K. Bong, D. Shin, J. Lee, S. Choi, H.J Yoo,”An energy-efficient and scalable deep learning/inference processor with tetra-parallel MIMD architecture for big data applications,” IEEE transactions on biomedical circuits and systems, Vol. 9, No.6, pp.838-848, 2015.

[4] J.A Jerkins, ”Motivating a market or regulatory solution to IoT insecurity with the Mirai botnet code,” IEEE Computing and Communication Workshop and Conference (CCWC) , pp.1-5, 2017.

[5] C. Kolias, G. Kambourakis, A. Stavrou, J. Voas, “DDoS in the IoT: Mirai and other botnets,“ IEEE Computer, Vol. 50, No.7, pp.80-84, 2017.

[6] A.O Prokofiev, Y.S Smirnova, V.A Surov, ”A method to detect Internet of Things botnets,” In Young Researchers in Electrical and Electronic Engineering (EIConRus), IEEE Conference of Russian , pp. 105-108, 2018

[7] G. Perrone, M. Vecchio, P,R. Pecori, “The Day After Mirai: A Survey on MQTT Security Solutions after the Largest Cyber-attack Carried Out through an Army of IoT Devices,” Second International Conference on Internet of Things, Big Data and Security PP.246-253, 2017.

[8] J. Smith-perrone, J. Sims, ”Securing cloud, SDN and large data network environments from emerging DDoS attacks,” 7th IEEE International Conference on Cloud Computing, Data Science & Engineering-Confluence, pp. 466-469,2017.

[9] A. Stanciu,T.C Balan, C. Gerigan,S. Zamfir, “Securing the IoT gateway based on the hardware implementation of a multi pattern search algorithm,” IEEE Optimization of Electrical and Electronic Equipment (OPTIM) & Aegean Conference on Electrical Machines and Power Electronics (ACEMP), pp. 1001-1006, 2017

[10] Q. Yaseen, M. Aldwairi, Y. Jararweh, M.Al-Ayyoub, B. Gupta, “Collusion attacks mitigation in internet of things: a fog based model,” Multimedia Tools and Applications, pp.1-20, 2017.

[11] Y. Yilmaz, S. Uludag, ”Mitigating IoT-based Cyber attacks on the Smart Grid,” 16th IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 517-522, 2017.

[12] M.E Ahmed, H. Kim, M. Park, “Mitigating DNS query-based DDoS attacks with machine learning on software-defined networking,” In Military Communications Conference (MILCOM), pp. 11-16, 2017.

[13] M. Stevanovic, J.M Pedersen, “Machine learning for identifying botnet network traffic,” Networking and Security Section, Department of Electronic Systems, Aalborg University, Tech. Rep. 2013.

[14] T. Zhu, S. Dhelim, Z. Zhou, S. Yang, H. Ning, “An architecture for aggregating information from distributed data nodes for industrial internet of things,” Computers & Electrical Engineering, 58, pp.337-349, 2017.

[15] K. Angrishi, ”Turning internet of things (IoT) into internet of vulnerabilities (IoV)”, IoT botnets,” 2017.

[16] D. H. Summerville, K. M. Zach, Y. Chen, “Ultra-lightweight deep packet anomaly detection for Internet of Things devices,” IEEE 34th International Performance Computing and Communications Conference (IPCCC), 2015.

[17] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, D. Breitenbacher, A. Shabtai, Y. Elovici, ”N-BaIoT: Network-based Detection of IoT Botnet Attacks Using Deep Auto encoders,” IEEE Pervasive Computing Vol.17 , No.3, 2018.

[18] C. McDermott, F. Majdani, A. V Petrovski, “Botnet detection in the internet of things using deep learning approaches,” IEEE International Joint Conference on Neural Networks (IJCNN), 2018.

[19] Y. Meidan, M. Bohadana, A. Shabtai, J. David Guarnizo, M. Ochoa, N. Ole Tippenhauer, Y. Elovici, “ProfilIoT: A machine learning approach for IoT device identification based on network traffic analysis,” In Proceedings of the Symposium on Applied Computing, pp. 506-509, 2017.

[20] S. Homayoun, M. Ahmadzadeh, S. Hashemi, A. Dehghantanha, R. Khayami, “BoTShark: A deep learning approach for botnet traffic detection,” Cyber Threat Intelligence, pp. 137-153, 2018.

[21] F. Shaikh, E. Bou-Harb, J. Crichigno, N. Ghani, ”A Machine Learning Model for Classifying Unsolicited IoT Devices by Observing Network Telescopes,” IEEE International Wireless Communications and Mobile Computing Conference (IWCMC 2018) ,2018.

[22] N. An, A. Duff, G. Naik. M, M. Faloutsos, S. Weber, S. Mancoridis, “Behavioral anomaly detection of malware on home routers,” IEEE 12th International Conference on

Malicious and Unwanted Software (MALWARE), pp. 47-54, 2017.

[23] L. Mathur, M. Raheja, P. Ahlawat, “Botnet Detection via mining of network traffic flow,” International Conference on Computational Intelligence and Data Science (ICCIDS 2018), pp. 1668-1678, 2018.

[24] A. Bijalwan, N. Chand, E. Shubhakar Pilli, C. R. Krishna, ”Botnet Analysis using Ensemble Classifier,” Perspectives in Science , pp. 502—504, 2016.

[25] F. Villegas Alejandre, N. Cruz Cortes, E. Aguirre Anaya, “Feature selection to detect Botnet using machine learning algorithms,” IEEE International Conference on Electronics, Communications and Computers (CONIELECOMP), 2017.

[26] S. Miller, C.C.R Busby-Earle, “The Impact of Different Botnet Flow Feature Subsets on Prediction Accuracy Using Supervised and Unsupervised Learning Methods,” International Journal of Internet Technology and Secured Transactions ,Vol. 5, No.2 , pp. 474-485, 2016.

[27] C.Hammerschmidt, S.Marchal, R. State, S. Verwer, “Behavioral clustering of non-stationary IP flow record data,” 12th International Conference on Network and Service Management, CNSM 2016 and Workshops, 3rd International Workshop on Management of SDN and NFV, ManSDN/NFV 2016, and International Workshop on Green ICT and Smart Networking, GISN 2016, pp. 297–301,2016.

[28] G. Kirubavathi Venkatesh, R. Anitha Nadarajan, “HTTP botnet detection using adaptive learning rate multilayer feed-forward neural network,” IFIP International Workshop on Information Security Theory and Practice, Springer, Berlin, Heidelberg, pp. 38-48 ,2012.

[29] K. Singh , S. Chandra Guntuku, A. Thakur, C. Hota, “Big data analytics framework for peer-to-peer botnet detection using random forests”, Information Sciences, pp.488-497, 2014.

[30] V. Hugo Bezerr, V.Guiherme Turrisi daCosta, S. Barbon Junior,R. Sanches Miani, B. Bogaz Zarpelao, “IoTDS: A One-Class Classification Approach to Detect Botnets in Internet of Things Devices,” Sensors, Vol. 19, No.14, pp.3188-95, 2019.

[31] K. Saleh Aloufi, “6LoWPAN Stack Model Configuration for IoT Streaming Data Transmission over CoAP,” International Journal of Communication Networks and Information Security (IJCNIS), Vol. 11, No. 2, pp 304-311, 2019.