Scientific Journal Impact Factor (SJIF) : 1.711
International Journal of Modern Trends in Engineering and Research
ISSN (ONLINE) : 2349 - 9745 ISSN (PRINT) : 2393 - 8161

An Evaluation of Classification Methods in Network Packet Protocol status Model for KDD Cup 99 Dataset

Apurva Tiwari[1], Sanjiv Sharma[2]
[1,2] *Department of CSE & IT*
Madhav Institute of Technology & Science, Gwalior (India)

Abstract - Distributed Denial of Service (DDoS) attacks generate enormous packets by a large no of agents and can easily exhaust the computing and communication resources of a victim within a short period of time. This paper proposes a method for construction of Network Packet Protocol Status for Detection of DDoS attack by exploiting its architecture which consists of Feature Selection, Classifier Generation sections. This paper is based on procedures of DDoS attack and then selects variables based on these features and shows the experiments with KDD Cup 99 Data Set to evaluate the proposed method. The KDD Cup 99 dataset has been the point of attraction for many researchers in the field of intrusion detection from the last decade. Various Classification methods are performed and on the basis of analysis, the most efficient Classification technique, in various conditions, can be found. The results show that each phase of the attack scenario is partitioned well and DDoS attack can be detected.

Keywords: *DDoS Attack, Network Packet Protocol Status Model, Classification, KDD Cup 99 dataset.*

I. Introduction

With the rapid development of network technologies, security becomes one of the most important issues today. DDoS attacks make a victim to deny providing normal services in the Internet by flooding a great number of malignant traffic. Attackers do not use the security holes of a network-connected system but launch attacks against its availability. It is agreed that DDoS attack has become a major threat to the stability of the Internet [1]. In a DDoS attack, an attacker compromises a large number of network-connected hosts by exploiting network software suceptibility [2]. Then, attack software is installed on these systems through secured channels. A large number of the compromised hosts on which attack software is installed send useless packets toward a victim at the same time. The volume of pernicious traffic generated by such hosts is so high that a victim cannot afford it. Various Data Mining (Classification) techniques have been employed to find models that are better understandable by the data owner [3, 4].

There are several categories of attacks found in the literature. For establishing a connection, an attacker may follow the same steps e.g., establishing a connection from source IP to the target IP and sending data to the attack target [5]. In KDD Cup 99 dataset [6] different attacks have different connections, as some of the attacks have few network connections. There are different feature values for normal and attack connections in the packet header, and the packet contents can be used as signatures for DDoS attack detection.

II. Related Work

A denial of service (DoS) attack is a malicious attempt to make a server or a network resource unavailable to users, usually by temporarily interrupting or suspending the services of a host connected to the Internet. Resources targeted in a DoS attack can be a specific computer, a port or service on the targeted system, an entire network, a component of a given network any system component. DoS attacks may also target human-system communications or human-response systems.

Since 1999, KDD'99 [11] has been the most vastly used data set for the assessment of anomaly detection methods. This data set is prepared by Stolfo et al. [12] and is built based on the data captured in DARPA'98 IDS evaluation program [13]. DARPA'98 is about 4 gigabytes of compressed raw (binary) tcpdump data of 7 weeks of network traffic, which can be processed into about 5 million connection records, each with about 100 bytes. The two weeks of test data have around 2 million connection records. KDD training dataset consists of about 4,900,000 single connection vectors each of which contains 41 features and is labeled as either normal or an attack, with exactly one specific attack type.

Janhavi Kaskar et al. [7] proposed a System for Detection of Distributed Denial of Service (DDoS) Attacks using KDD Cup Data Set Distributed denial-of-service (DDoS) attacks are a major security threat, the prevention of which is very hard, like when it comes to highly distributed daemon-based attacks. The early discovery of these attacks, although difficult, is necessary to protect network resources as well as the end users. This paper addresses the problem of DDoS attacks and present the foundation and algorithms of IDS. The base of system is composed of intrusion detection systems (IDSs) which use the KDD Cup dataset to detect intrusion. The IDS scans all the files being transmitted from the routers for malicious content and known virus signatures. The evaluation of this system, using the KDD testing dataset, shows a better ratio of detecting attacks and a low false positives ratio. It also supports easy modifiability, scalability and usability.

Mohammad Khubeb Siddiqui et al.[8] proposed an Analysis of KDD CUP 99 Dataset using Clustering based Data Mining. Many researchers have contributed their efforts to analyze the dataset by different techniques. Analysis can be used in any type of industry that produces and consumes data that includes security. This paper is an analysis of 10% of KDD cup'99 training dataset based on intrusion detection and focuses on establishing a relationship between the attack types and the protocol used by the hackers, using clustered data. Analysis of data is performed using k-means clustering; These authors used the Oracle 10g data miner as a tool for the analysis of dataset and build 1000 clusters to segment the 494,020 records. The investigation revealed many interesting results about the protocols and attack types preferred by the hackers for intruding the networks.

Keunsoo Lee et al.[9] proposed DDoS attack detection method using cluster analysis. Distributed Denial of Service (DDoS) attacks generate enormous packets by a large number of agents and can easily exhaust the computing and communication resources of a victim within a short period of time. This paper proposes a method for proactive detection of DDoS attack by exploiting its architecture which consists of the selection of handlers and agents, the communication and compromise, and attack. The procedures of DDoS attack are described and then variables based on these features are selected. After that, authors perform cluster analysis for proactive detection of the attack. Authors experiment with 2000 DARPA Intrusion Detection Scenario Specific Data Set in order to evaluate the method. The results show that each phase of the attack scenario is partitioned well and anyone can detect precursors of DDoS attack as well as the attack itself.

Rui Zhong et al.[10] proposed DDoS Detection System Based on Data Mining. Distributed Denial of Service Attack (DDoS) brings a very serious threat to send to the stability of the Internet This paper analyzes the characteristic of the DDoS attack and recently DDoS attack detection method. Presents a DDoS attack detection model based on Data Mining Algorithm. FCM cluster algorithm and Apriori association algorithm used to extract network traffic model and network packet protocol status model. The threshold is set for detection model. Experimental result shows that DDoS attacks can be detected efficiently and swiftly.

Mihui Kim et al. [19] proposed a combined Data Mining approach for DDoS Attack Detection. Recently, as the serious damage caused by DDoS attacks increases, the rapid detection and the proper response mechanisms are urgent. Existing security mechanisms do not provide adequate defense against these attacks, or the defense proficiency of some mechanisms is finite to specific DDoS attacks. It is indispensable to analyze the fundamental features of DDoS attacks because these attacks can easily diversify the used port/protocol, or operation method. This paper proposes a combined data mining approach for modeling the traffic pattern of normal and diverse attacks. This

approach uses the automatic feature selection mechanism for selecting the relevant attributes. And the classifier is built with the theoretically selected attribute through the neural network. And then, the experimental results show that our approach can provide the best performance on the real network, in comparison with that by heuristic feature selection and any other single data mining approaches.

Category	Attack type
Normal	Normal(97277)
DoS	Back(2203), Land(21), Neptune(107201), Pod(264), Smurf(280790), Teardrop(979)
U2R	Buffer_overflow(30), loadmodule(9), perl(3), Rootkit(10)
R2L	ftp_write(8), Guess_passwd(53), Imap(12), Multihop(7), Phf(4), Spy(2), Warezclient(1020), Warezmaster(20)
Probe	Ipsweep(1247), Nmap(231), Portsweep(1040), Satan(1589)

Table 1: Attack Types and Size in 10% KDD Data Set

III. Proposed Model

DDoS attack launched by the attacker includes mainly three steps, that is, searching the attack target, attacking and occupying the vulnerable nodes and actual attacks. The specific process is as follows:
Before attacking, the attacker firstly searches the hosts in the network with security vulnerabilities from which the hosts with good link state and performance are picked out, and then intrudes these hosts so that corresponding administration authority is achieved to install control programs.

1. The attacker through network gives the handlers of the attack control instructions that cause the handlers give orders to the agents. Generally the attack agents are controlled by more than one handlers.
2. The agents send the victims a large quantity of packets. It is difficult to distinguish between malicious requests and normal connection requests because these packets are masqueraded and could not be recognized where they are from as well as the protocols used by attackers are very common.

The characteristics of DDoS Attack:

The characteristics of DDoS Attack are as follows after the analysis of it:

1. Abnormal traffic. A lot of useless packets transmitted by the attacker in order to occupy the resources of the victims (bandwidth or host resources). Such a large number of packets would cause the victims system-halted and fail to provide external services.
2. Most DDoS attacks take the three times handshake mechanism and use “SYN” status flag to send the victim connection requests . However, this does not mean to build a real connection, which makes the victim maintain a great deal of half-opened connection and consume the resources of the victims.
3. The attacker makes use of one of the characters of TCP/IP protocol that some non-compliant packets could be used so as to launch DDoS attacks.

Architecture of Proposed Methodolgy in shown in fig 1 :

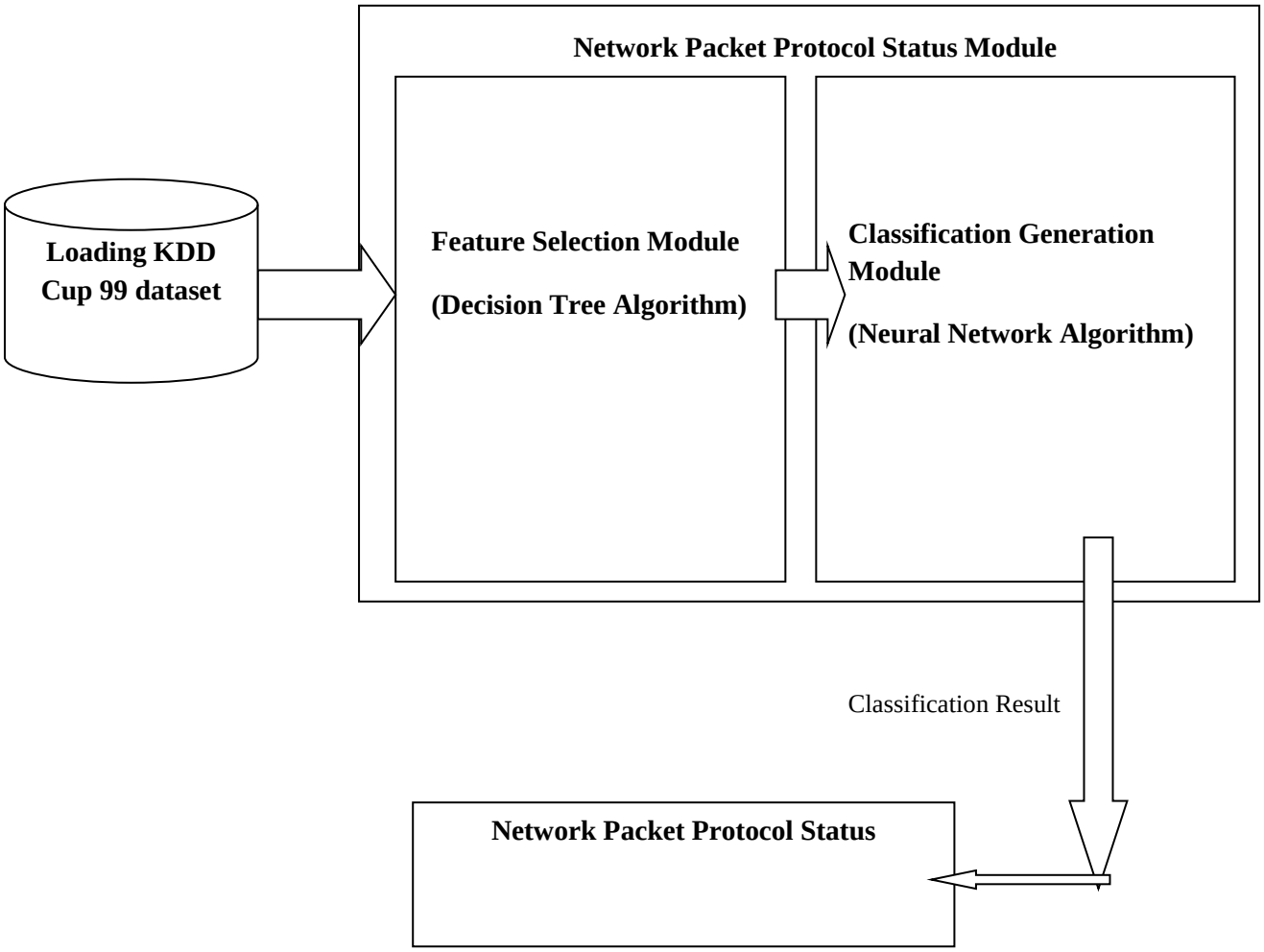


Fig 1: Architecture of Proposed Methodology

Proposed Model is shown in Fig 1.

Establishment of Network Packet Protocol Status Model:

DDoS Attack Procedure:
Fig 1 is the working traffic chart of the proposed model that combines the abnormal traffic analysis and the network packet protocol status detection model. The specific procedure is as follows:
1. KDD Cup 99 Data Set is loaded in the model.
2. This data set is sent to Network Packet Protocol Status Module.
3. Feature Selection Module selects the appropriate features from this data set using Decision Tree Algorithm.
4. Now, this specific data set is sent to Classifier Generation Module.
5. Neural Network Algorithm is applied on this data set for Classifier Generation.
6. The result of Classifier Generation is sent to the Network Packet Protocol Status and this result presents Network Packet Protocol Status.

Classification Parameters:

1. Correctly Classified Instances: All the instances must be classified correctly, i.e. 100%.

2. Incorrectly Classified Instances: No instance must be incorrectly classified.

3. Kappa Statistics: The Kappa statistic (K) is a metric that compares an Observed Accuracy with an Expected Accuracy.

$$\mathbf{K = (P_r(a) - P_r(e))/(1 - P_r(e))} \quad \mathbf{(3.1)}$$

Where $P_r(a)$ is the relative observed agreement among raters, $P_r(e)$ is the hypothetical probability of chance agreement. If the raters are in complete agreement then k = 1.

4. Mean Absolute Error: MAE is a quantity used to measure how close forecasts or predictions are to the eventual outcomes. The mean absolute error is given by

$$\mathbf{MAE = \frac{1}{n}\sum_{i=1}^{n}|f_i - y_i| = \frac{1}{n}\sum_{i=1}^{n} e_i} \quad \mathbf{(3.2)}$$

Where f_i is the prediction and $\mathbf{y_i}$ is the true value.

5. Root Mean Squared Error: is a frequently used measure of the difference between values predicted by a model and the values actually observed from the environment that is being modelled.

$$\mathbf{RMSE = \sqrt{\frac{\sum_{i=1}^{n}(X_{obs,i} - X_{model,i})^2}{n}}} \quad \mathbf{(3.3)}$$

where X_{obs} is observed values and X_{model} is modelled values at time/place i.

6. Relative Absolute Error: the relative absolute error (E_i) takes the total absolute error and normalizes it by dividing by the total absolute error of the simple predictor.

$$\mathbf{E_i = \frac{\sum_{j=1}^{n}|P_{ij} - T_j|}{\sum_{j=1}^{n}|T_j - \bar{T}|}} \quad \mathbf{(3.4)}$$

where P_{ij} is the value predicted by the individual program *i* for sample case j (out of n sample cases), T_j is the target value for sample case j, $\bar{T}$ is given by the formula:

$$\mathbf{\bar{T} = \frac{1}{n}\sum_{i=1}^{n} T_i} \quad \mathbf{(3.5)}$$

7. Root Relative Squared Error: RRSE is relative to what it would have been if a simple predictor had been used. More specifically, this simple predictor is just the average of the actual values.

The RRSE (E_i) of an individual program i is evaluated by the equation:

$$\mathbf{E_i} = \sqrt{\frac{\sum_{j=1}^{n}(P_{ij}-T_j)^2}{\sum_{j=1}^{n}(T_j-\bar{T})^2}} \quad \textbf{(3.6)}$$

where P_{ij} is the value predicted by the individual program i for sample case j (out of n sample cases), T_j is the target value for sample case j.

8. Time Taken: It is the time to complete the classification process. Time taken must be minimum as much as possible.

Confusion Matrix Parameters: also known as a contingency table or an error matrix.

1. TP Rate: refers to positive instances that correctly labeled the classifier (When abnormal data detected as abnormal).

$$\mathbf{TPR} = \frac{\mathbf{TP}}{(\mathbf{TP+FN})} \quad \textbf{(3.7)}$$

Where TP is True Positive, FN is False Negative.

2. FP Rate: is the negative instances that were incorrectly labeled (when normal data detected as abnormal).

$$\mathbf{FPR} = \frac{\mathbf{FP}}{(\mathbf{FP+TN})} \quad \textbf{(3.8)}$$

Where FP is false Positive and TN is True Negative.

3. Precision: Precision (P) is the fraction of retrieved instances that are relevant.

$$\mathbf{P} = \frac{\mathbf{TP}}{(\mathbf{TP+FP})} \quad \textbf{(3.9)}$$

4. Recall: Recall (R) (or Sensitivity) is the fraction of relevant instances that are retrieved.

$$\mathbf{R} = \frac{\mathbf{TP}}{(\mathbf{TP+FN})} \quad \textbf{(3.10)}$$

5. F- Measure: The F-measure is defined as a harmonic mean of precision (P) and recall (R).

$$\mathbf{F} = \frac{\mathbf{2PR}}{(\mathbf{P+R})} \quad \textbf{(3.11)}$$

6. ROC Area: A receiver operating characteristic (ROC), or ROC curve, is a graphical plot that illustrates the performance of a binary classifier system as its discrimination threshold is varied.

Feature Selection Module:

Due to the large amount of data flowing over the network real time intrusion detection is almost impossible. Feature selection can reduce the computation time and model complexity. Research on feature selection started in early 60s [15]. Basically feature selection is a technique of selecting a subset of relevant/important features by removing most irrelevant and redundant features [16] from the data for building an effective and efficient learning model [17]. Selection processes involve four basic steps in a typical feature selection method [18] shown in Figure. First is generation procedure to generate the next candidate subset; second one is an evaluation function to evaluate the subset and third one is a stopping criterion to decide when to stop; and a validation procedure to check whether the subset is valid.

This paper analyses four Decision Tree Algorithms (Decision Stump, J48, Random Forest, CART) for Feature Selection Module. Classification Parameters and Confusion Matrices are described in the tables.

Decision Stump: A Decision Stump is a machine learning model consisting of a one-level decision tree [20]. That is, it is a decision tree with one internal node (the root) which is immediately connected to the terminal nodes (its leaves). A decision stump makes a prediction based on the value of just a single input feature.

Classification Parameters : Decision Stump classification parameters are given in the Table 2:

Correctly classified instances	172 (80.3738 %)
Incorrectly classified instances	42 (19.6262 %)
Kappa statistic	0.674
Mean absolute error	0.1645
Root mean squared error	0.2868
Relative absolute error	39.202 %
Root relative squared error	62.6483 %
Total Number of Instances	214

Table 2: classification Parameters of Decision Stump

Confusion Matrix : Confusion Matrix of Decision Stump is shown in Table 3:

	TP Rate	**FP Rate**	**Precision**	**Recall**	**F Measure**	**ROC Area**	**Class**
	1	0.294	0.628	1	0.772	0.846	Tcp
	0	0	0	0	0	0.783	Udp
	1	0	1	1	1	1	Icmp
Weighted Average	0.804	0.097	0.68	0.804	0.728	0.906	

Table 3: Confusion Matrix of Decision Stump

J48: J48 classifier is a simple C4.5 decision tree for classification. It creates a binary tree. The decision tree approach is most useful in classification problem. With this technique, a tree is constructed to model the classification process. Once the tree is built, it is applied to each tuple in the database and results in classification for that tuple.

Classification Parameters: J48 Classification Parameters are given in Table 4:

Correctly classified instances	212 (99.0654 %)
Incorrectly classified instances	2 (0.9346 %)
Kappa statistic	0.9851
Mean absolute error	0.0093
Root mean squared error	0.0792
Relative absolute error	2.2062 %
Root relative squared error	17.3054 %
Total Number of Instances	214
Time taken	0.06 sec

Table 4: Classification Parameters of J48

Confusion Matrix: Confusion Matrix of J48 classification is given in Table 5:

	TP Rate	FP Rate	Precision	Recall	F Measure	ROC Area	Class
	0.986	0.007	0.986	0.986	0.986	0.986	Tcp
	1	0	1	1	1	1	Udp
	0.99	0.009	0.99	0.99	0.99	0.988	Icmp
Weighted Average	0.991	0.006	0.991	0.991	0.991	0.99	

Table 5: Confusion Matrix of J48

Random Forest. Random Forests are an ensemble learning method for classification and regression that construct a number of decision trees at training time and outputting the class that is the mode of the classes output by individual trees

Classification Parameters: Random Forest Classification Parameters are shown in the Table 6:

Correctly classified instances	214 (100 %)
Incorrectly classified instances	0 (0%)
Kappa statistic	1
Mean absolute error	0
Root mean squared error	0
Relative absolute error	0 %
Root relative squared error	0 %
Total Number of Instances	214
Time taken	0.31 sec

Table 6 : classification Parameters of Random Forest

Confusion Matrix: Confusion Matrix of Random Forest is given in Table 7:

	TP Rate	FP Rate	Precision	Recall	F Measure	ROC Area	Class
	1	0	1	1	1	1	Tcp
	1	0	1	1	1	1	Udp
	1	0	1	1	1	1	Icmp
Weighted Average	1	0	1	1	1	1	

Table 7 : Confusion Matrix of Random Forest

CART: The CART (Classification & Regression Tree) algorithm is structured as a sequence of questions, the answers to which determine what the next question, if any should be. The result of these questions is a tree like structure where the ends are terminal nodes at which point there are no more questions.

Classification Parameters: Classification Parameters of CART Algorithm is given in Table 8:

Correctly classified instances	214 (100 %)
Incorrectly classified instances	0 (0 %)
Kappa statistic	1
Mean absolute error	0.0087
Root mean squared error	0.0436
Relative absolute error	2.0658 %
Root relative squared error	9.5331 %
Total Number of Instances	214
Time taken	0.09 sec

Table 8: Classification Parameters of CART

Confusion Matrix: Confusion Matrix of CART given in Table 9:

	TP Rate	**FP Rate**	**Precision**	**Recall**	**F Measure**	**ROC Area**	**Class**
	1	0	1	1	1	1	Tcp
	1	0	1	1	1	1	Udp
	1	0	1	1	1	1	Icmp
Weighted Average	1	0	1	1	1	1	

Table 9: Confusion Matrix of CART

Classifier Gene

Classification is a Supervised Learning Approach. Data classification deals with the process of finding the common properties among a set of objects in a database and divides them into different categories. The basic idea of classification techniques is to use some limited set of records, named a training set Classification is a form of data analysis that extracts models describing important data classes. Such models, called classifiers, predict categorical (discrete, unordered) class labels. For example,we can build a classification model to categorize bank loan applications as either safe or risky.Such analysis can help provide us with a better understanding of the data at large. There are various classifiers available in WEKA. This paper compares some results of those classifiers.

Neural Network is used as a tool for classification. The idea of neural computing grew out of a desire to capture the pattern recognition capabilities of a biological brain. Muculloch-Pitts developed the first model of a physiological brain called 'Muculloch-Pitts neuron', which became the basis for almost all the artificial neural networks, where nodes are likened to neurons and arcs to dendrites or axons. The neural networks were developed for recognition of ill-defined objects such as hand written characters, finger prints, speech, electricals or sonar signals, and double spirals. They have also been used for detection of faults in a chemical process, explosives in airline baggage, and prediction of bank failures. Neural network models are non-parametric and are able to adjust the form of the discrimination function to fit the data.

This paper analyses two Neural Network Algorithms (Multilayer Perceptron, Radial Basis Function) for Classifier Generation Module. Classification Parameters and Confusion Matrices are described in the tables:

Multilayer Perceptron: MLP is a finite directed acyclic graph. Nodes that are no target of any connection are called input neurons. Nodes that are no source of any connection are called output

neurons. All nodes that are neither input neurons nor output neurons are called hidden neurons. Since the graph is acyclic, all neurons can be organized in layers, with the set of input layers being the first layer.

Classification Parameters: Table 10 shows the classification parameters result for Multilayer Perceptron Algorithm.

Correctly classified instances	213 (99.5327 %)
Incorrectly classified instances	1 (0.4673 %)
Kappa statistic	0.9926
Mean absolute error	0.0045
Root mean squared error	0.0561
Relative absolute error	1.0725 %
Root relative squared error	12.2507 %
Total Number of Instances	214
Time taken	0.37 sec

Table 10: Classification Parameters of Multilayer Perceptron

Confusion Matrix: Table 11 displays confusion matrix of Multilayer Perceptron.

	TP Rate	**FP Rate**	**Precision**	**Recall**	**F Measure**	**ROC**	**Class**
	1	0	1	1	1	1	TCP
	1	0	1	1	1	1	UDP
	1	0	1	1	1	1	ICMP
Weighted Average	1	0	1	1	1	1	

Table 11: Confusion Matrix of multilayer Perceptron

Radial Basis Function: A radial basis function (RBF) is a real-valued function whose value depends only on the distance from the origin.

Classification Parameters: Table 12 shows classification parameters of Radial Basis Function.

Correctly classified instances	214 (100 %)
Incorrectly classified instances	0 (0 %)
Kappa statistic	1
Mean absolute error	0.0053
Root mean squared error	0.0289
Relative absolute error	1.2555 %
Root relative squared error	6.3098 %

Total Number of Instances	214
Time taken	7.66 sec

Confusion Mat ction algorithm.

Table 12 : Classification Parameters of Radial Basis Function

	TP Rate	FP Rate	Precision	Recall	F Measure	ROC Area	Class
	1	0.007	0.986	1	0.993	0.995	TCP
	1	0	1	1	1	1	UDP
	0.99	0	1	0.99	0.995	0.997	ICMP
Weighted Average	0.995	0.002	0.995	0.995	0.995	0.997	

Table 13: Confusion Matrix of Radial Basis Function

IV. Analysis:

Comparison Among Various Feature Selection Algorithms:

This paper is comparing four Feature Generation Classification Algorithms on the basis of various Classification Parameters. Comparison can be easily understood by plotting graphs. Comparison graphs are given in Fig 2:

Correctly Classified Instances:

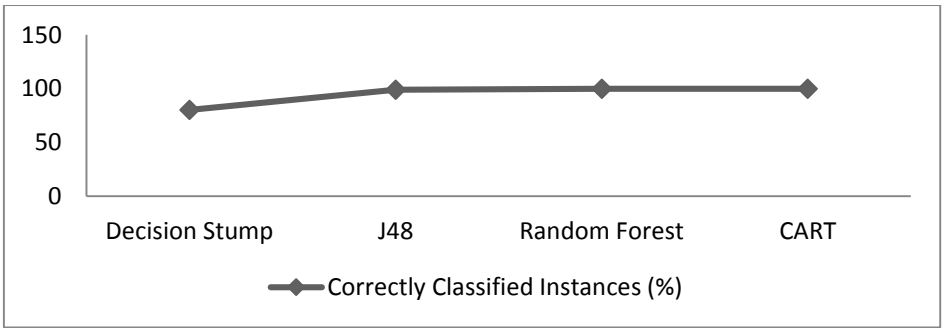


Fig 2: Correctly classified instances

Figure shows that Random Forest and CART Algorithms classify instances 100%, J48 provides near 100% classification.

Kappa Statistics: The Kappa statistic (or value) is a metric that compares an Observed Accuracy with an Expected Accuracy (random chance).

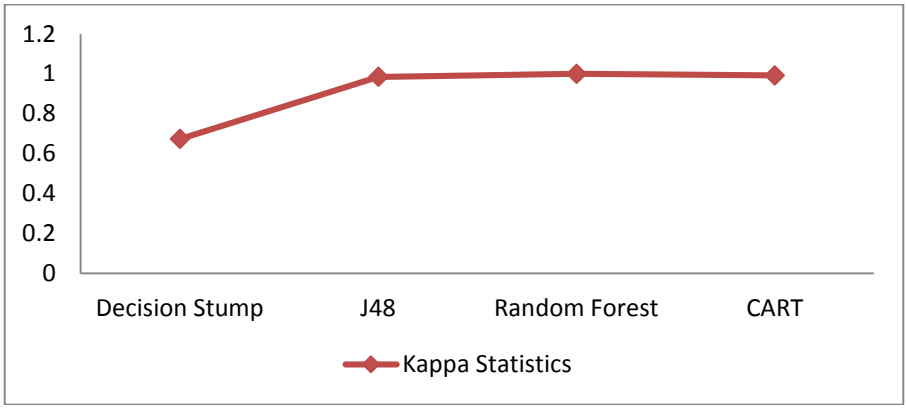


Fig 3: Kappa Statistics

Fig 3 shows that Random Forest and CART give ideal result (i.e. 1) for Kappa Statistics. J48 outputs the value near 1.

Mean Absolute Error: The mean absolute error (MAE) is a quantity used to measure how close forecasts or predictions are to the eventual outcomes.

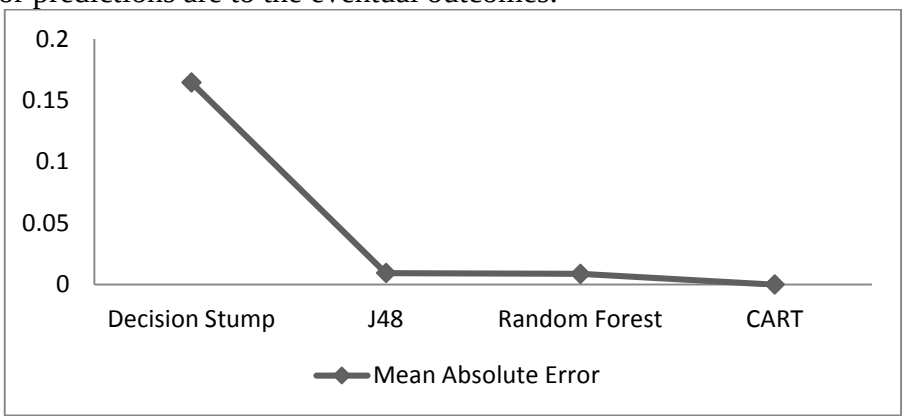


Fig 4: Mean Absolute Error

Value of Mean Absolute Error must be 0. In the Fig 4, It can be observed that among the four algorithms, CART gives the best result.

Root Mean Squared Error: RMSE is a frequently used measure of the difference between values predicted by a model and the values actually observed from the environment that is being modelled.

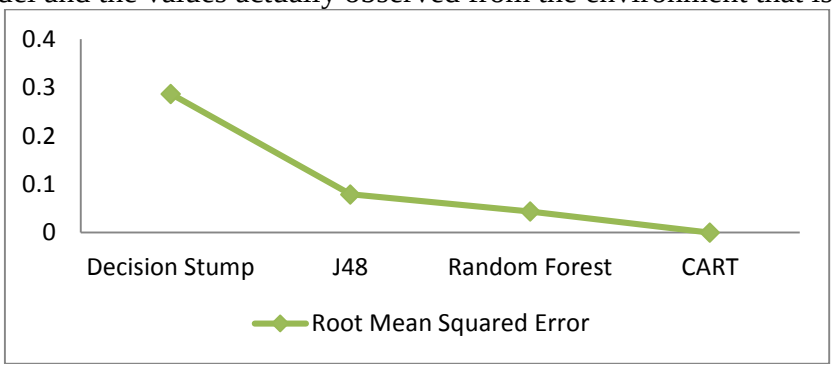


Fig 5: Root Mean Squared Error

Root Mean Squared Error must be 0 ideally. From the Fig 5, CART gives the ideal result , i.e. 0.

Relative Absolute Error: It is very similar to the relative squared error in the sense that it is also relative to a simple predictor, which is just the average of the actual values.

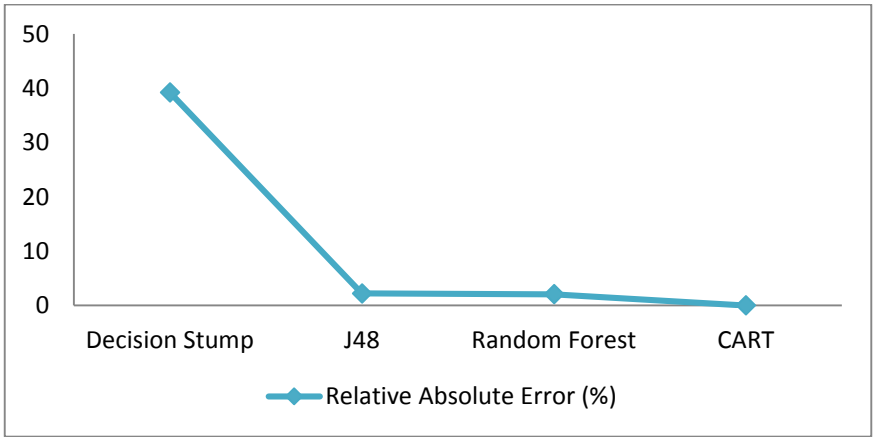


Fig 6: Relative Aboslute Error (%)

In ideal case, Relative Absolute Error must be 0 %. From the Fig 6, it can be observed that CART gives the ideal result among four algorithms.

Root Relative Squared Error: It is relative to what it would have been if a simple predictor had been used.

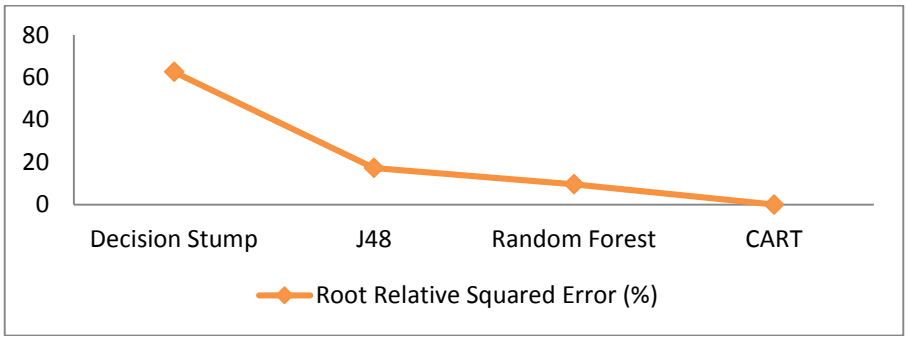


Fig 7: Root Relative Squared Error (%)

From the Fig 7, it can be seen that CART gives the desired result for Root Relative Squared Error. Decision Stump gives very much erroneous result.

Time Taken: It is the time taken to complete the classification process.

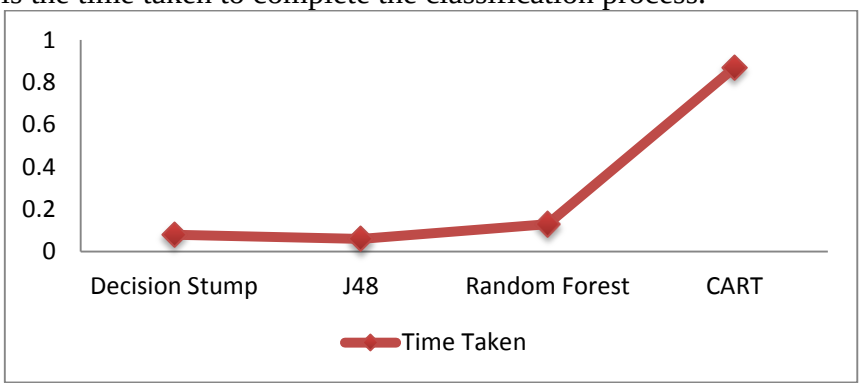
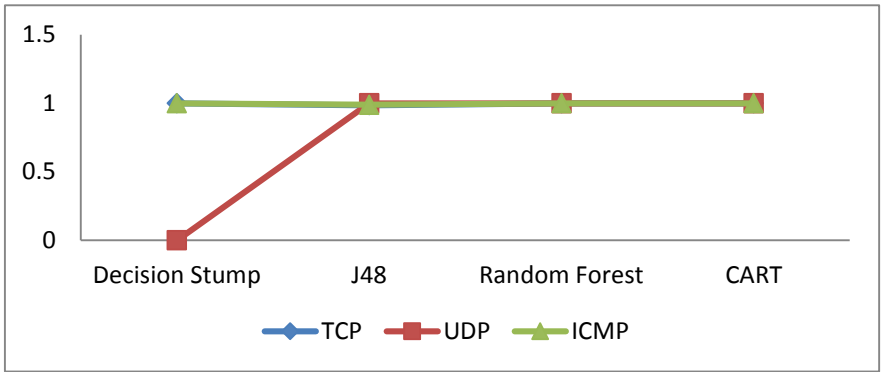


Fig 8: Time Taken

From the Fig 8, it can be observed that CART algorithm takes more time than that of any other algorithm. Difference between the time taken by any two of the three algorithms (Decision Stump, J48, Random Forest) is not very much. So, any one of the three can be chosen on the basis of Time Taken.

Comparison Among Confusion Matrix Parameters:

TP Rate:



Ideally True Positive Rate must be 1. For TCP protocol, J48, Random Forest and CART give ideal answer. For UDP protocol, Random Forest and CART provide ideal result. For ICMP traffic, Decision Stump, Random Forest and CART produce ideal result. Thus, it can be said that Random Forest and CART produce best result for the three types of traffic.

FP Rate:

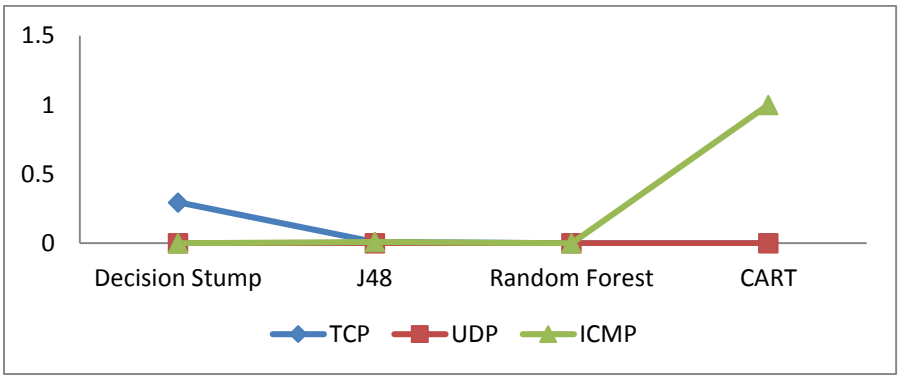


Fig 10 : FP Rate

False Positive (FP) Rate must be 0 for all kind of traffic. Random Forest and CART algorithms produces best results for FP Rate.

Precision:

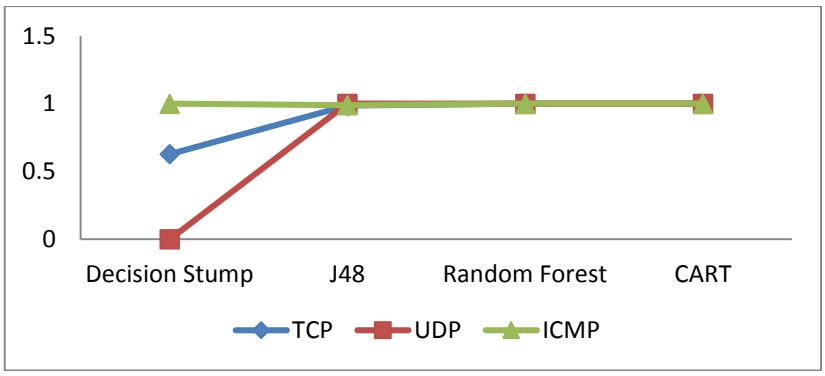


Fig 11: Precision

Precision or Positive Predictive Value must be 1. For TCP traffic, Random Forest and CART produce best results. In case of UDP traffic, three algorithms except Decision Stump provide ideal results. For ICMP traffic, three mechanism except J48 outputs ideal results.

Recall:

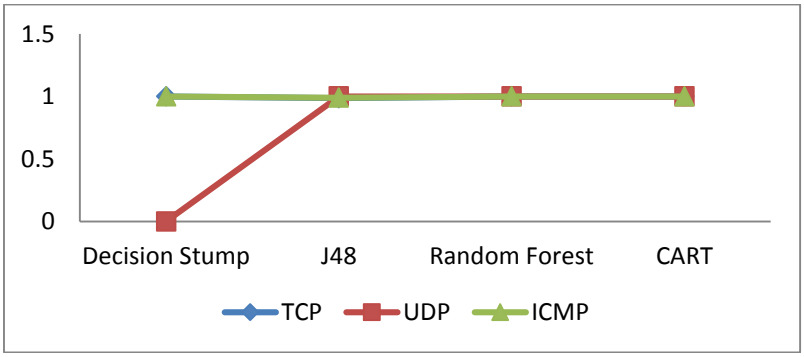


Fig 12: Recall

Optimal value of Recall or Sensitivity is 1. For the three types of traffic, Random Forest and CART produce the optimal value.

F Measure:

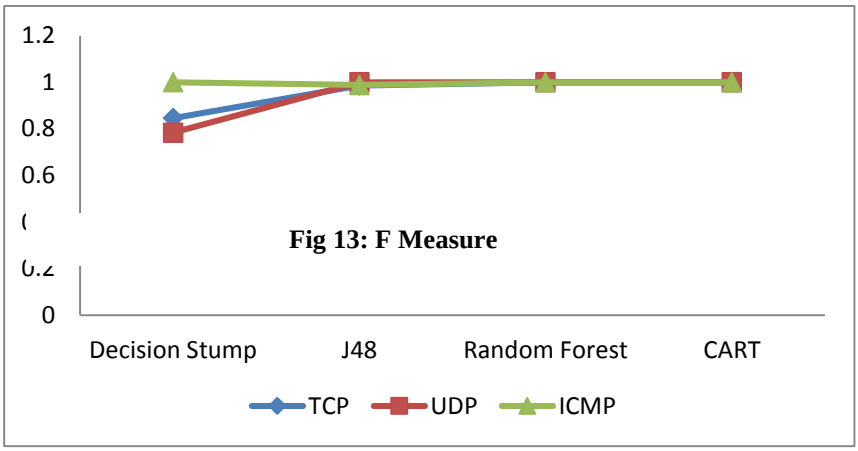


Fig 13: F Measure

F Measure is the harominc mean of Precision and Recall. Random Forest and CART produce the optimal result for three types of traffic.

ROC Area:

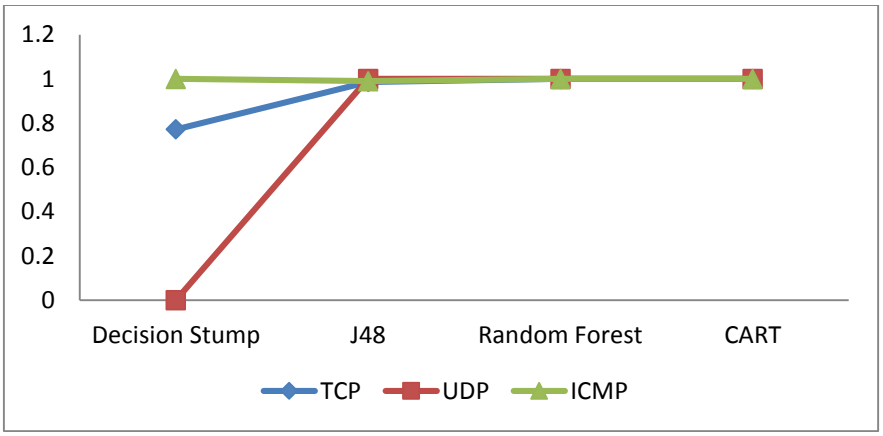


Fig 14: ROC Area

If Receiver Operating Characteristic (ROC) curve presents an area of 1, then it is a perfect test. For, TCP, UDP, ICMP protocols, Random Forest and CART poduce the perfect results.

Comparison Among Various Neural Network Algorithms: This paper is comparing two Classifier Generation Neural Network Algorithms on the basis of various Classification Parameters. Comparison can be easily understood by plotting graphs. Comparison graphs are given in Fig 14:

Correctly Classified Instances:

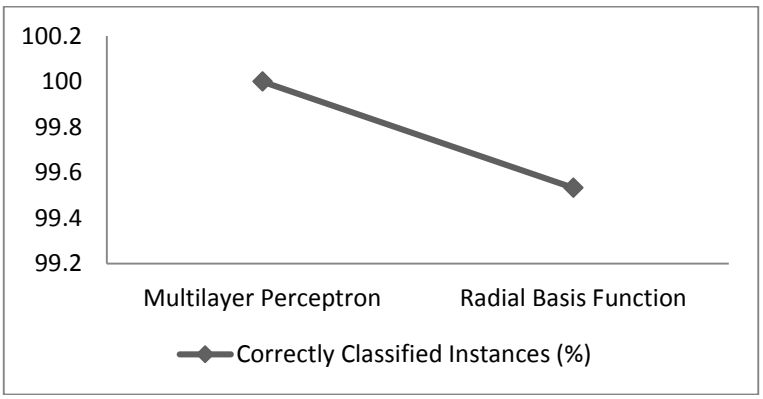


Fig 14: Correctly Classified Instances

Fig 14 shows that Multilayer Perceptron 100 % classifies the instances thus gives better result than that of Radial Basis Function.

Kappa Statistics:

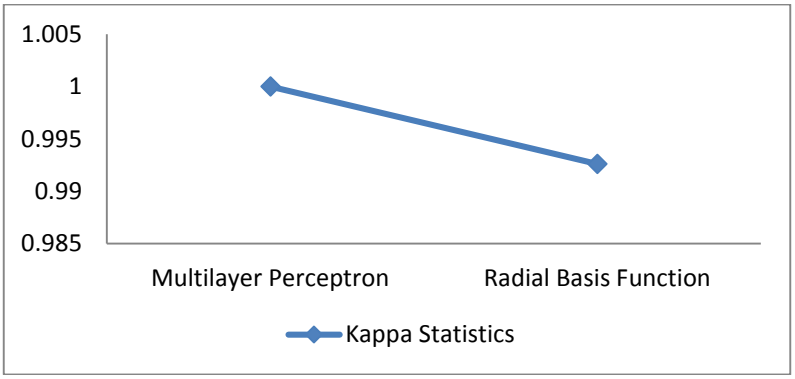


Fig 15: Kappa Statistics

In Fig 15, it can be observed that Multilayer perceptron produces optimal answer for Kappa Statistics (i.e. 1).

Mean Absolute Error:

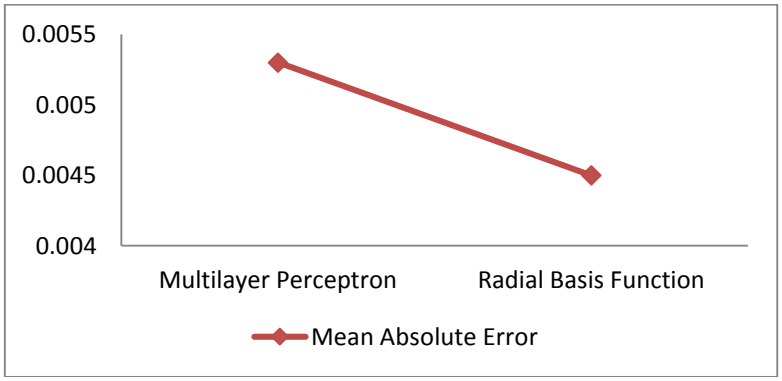


Fig 16: Mean Absolute Error

it can be analyzed by Fig 16 that Radial Basis Function produces less error (Mean Absolute Error) than that of Multilayer Perceptron.

Root Mean Squared Error:

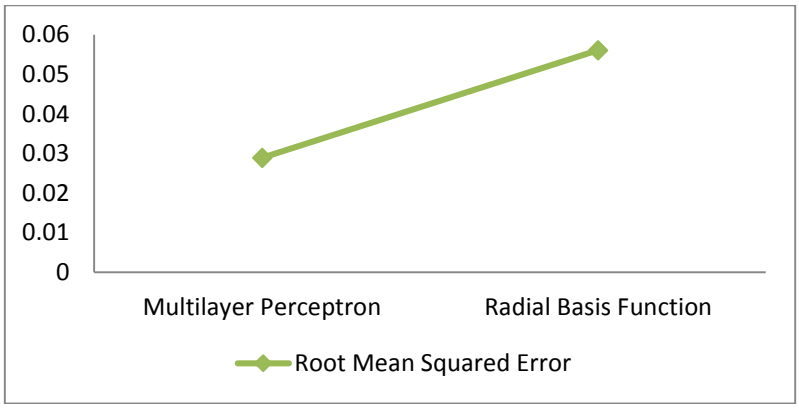


Fig 17: Root Mean Squared Error

From Fig 17, Root Mean Squared Error value for Multilayer Perceptron is less than that for Radial Basis Function, thus Multilayer Perceptron produces better result.

Relative Absolute Error:

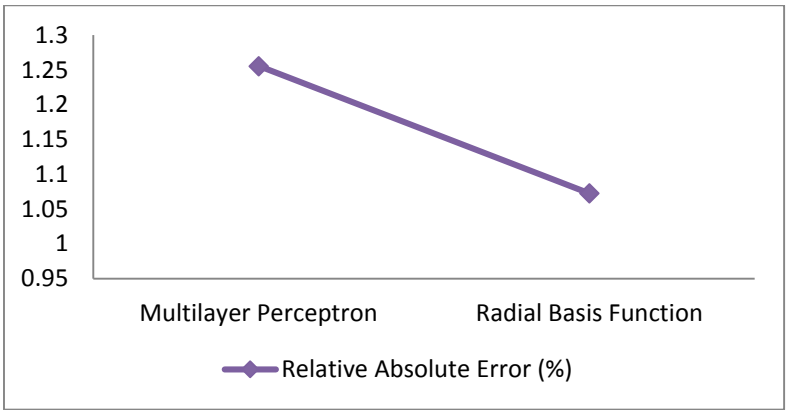


Fig 18: Relative Absolute Error (%)

From fig 18, Radial Basis Function produces better result for Relative Absolute Error (1.0725 %) than that for Multilayer Perceptron (1.2555 %).

Root Relative Squared Error:

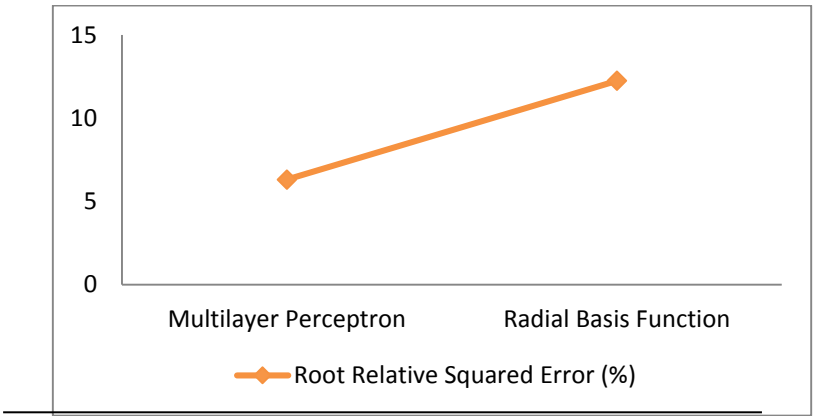


Fig 19: Root Relative Squared Error (%)

Fig 19 displays that Multilayer Perceptron produces better result (6.3098 %) than that of Radial Bais Function (12.2507 %).

Time Taken:

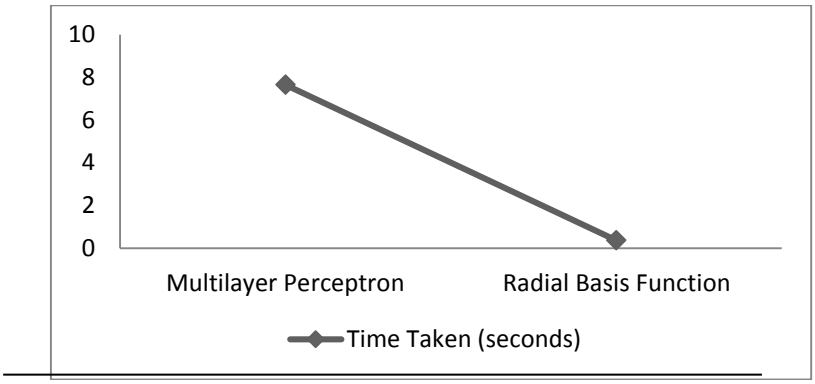


Fig 20: Time Taken (seconds)

From Fig 20, Radial Basis Function takes less time (0.37 sec) than that of Multilayer Perceptron (7.66 sec).

Confusion Matrix Parameters:

TP Rate:

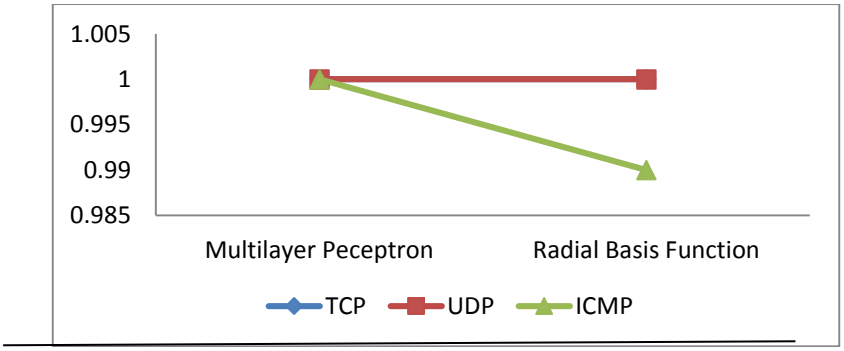


Fig 21: TP Rate

Fig 21 refers to positive instances that correctly labeled the classifier.

FP Rate:

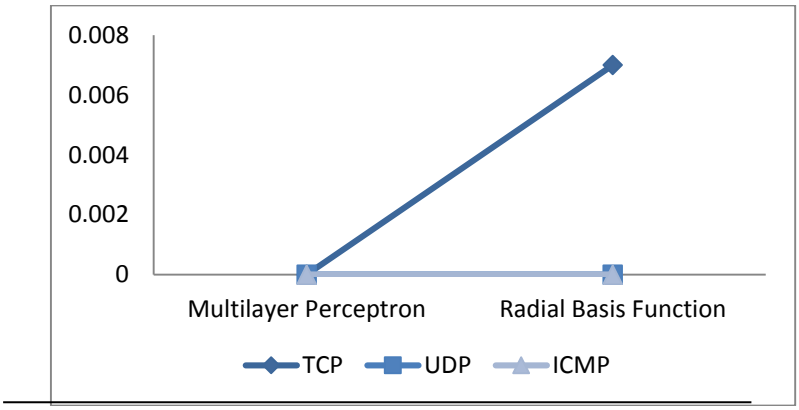


Fig 22: FP Rate

Fig 22 shows that False Positive (FP) Rate (negative instances that were incorrectly labeled) is optimal for Multilayer Perceptron. Radial Basis Function produces optimal result for UDP and ICMP traffic.

Precision:

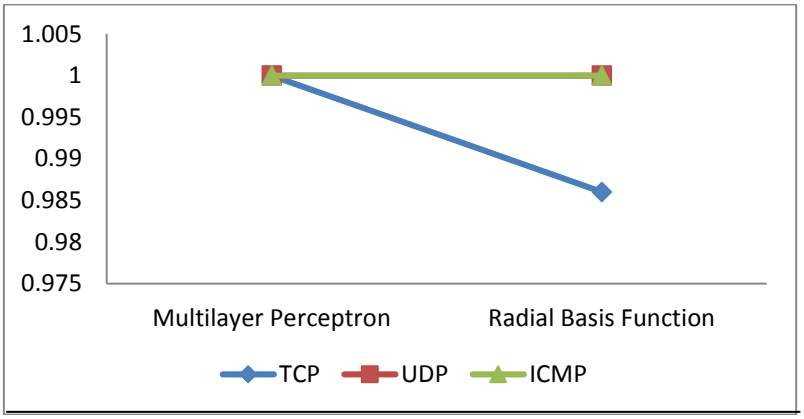


Fig 23: Precision

Fig 23 shows Precision (fraction of retrieved instances that are relevant). Multilayer Perceptron produces optimal results.

Recall:

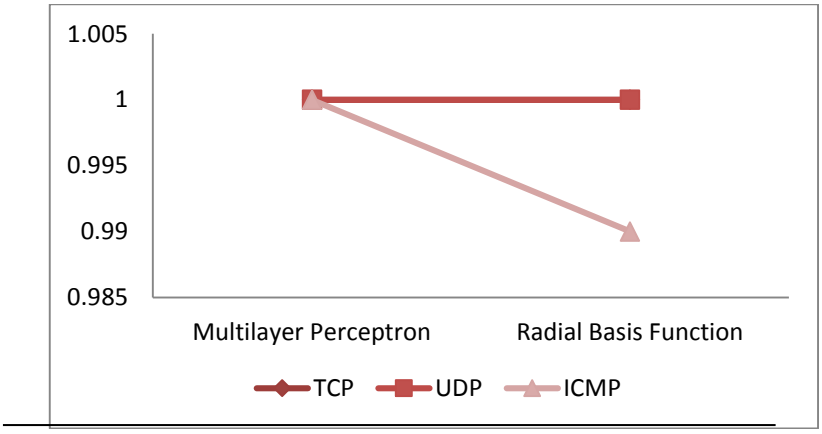


Fig 24: Recall

Fig 24 shows Recall (fraction of relevant instances that are retrieved). Here again, Multilayer Perceptron gives the best results.

F Measure:

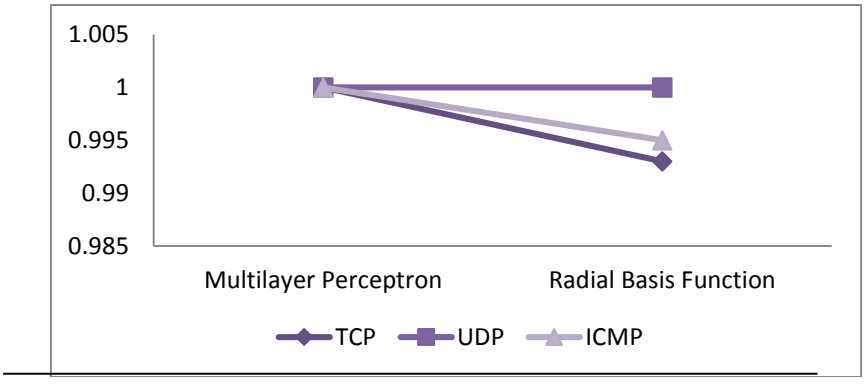


Fig 25: F Measure

Fig 25 shows F Measure (mean of precision and recall). Multilayer perceptron produces optimal result for TCP, UDP and ICMP traffic.

ROC Area:

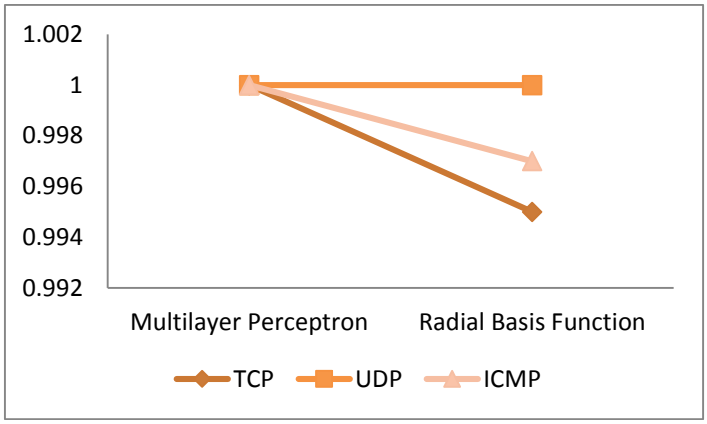


Fig 26: ROC Area

Fig 26 shows ROC area for Multilayer Perceptron (TCP, UDP and ICMP traffic) produces prefect result.

V. CONCLUSION & FUTURE WORK

This paper shows experiment and analysis of KDD Cup 99 data set and represents Network Packet Protocol Status Model for detection of DDoS attack. In this paper, four feature selection algorithms are compared on the basis of Classification Parameters and Confusion Matrix. It can be observed that among the four algorithms, CART is the only algorithm, that always produces the optimal result for each and every parameter except Time Taken. From the comparison graphs, it is clear that eventhough Time Taken by CART to complete the process is more than any other algorithm, but it is not too much to ignore this mechanism. Thus, CART provides optimal result for feature selection process.

In case of Classifier Generation phase, there are two methods to be compared: Multilayer Perceptron (MLP) and Radial Basis Function (RBF). Multilayer perceptron model classifies all the instances correctly, but Radial Basis Function does not. On the basis of Confusion Matrix , Multilayer Perceptron gives the optimal matrix for all the parameters. But the problem with MLP is that, it takes more time to run than that of RBF and this gap cannot be ignored.

This research provides a scope for analysis of KDD cup 99 dataset and modification of existing DDoS detection system. This paper focuses only on Network Packet Protocol Status Model. Furthermore, for improvement of DDoS detection model, traffic threshold model and detection of anomaly in network traffic is required.

REFERENCES

1. Computer Emergency Response Team (1999). Results of the distributedsystems intruder tools workshop. <http://www.cert.org/reports/ dsit_workshop-final.html>.
2. Xu, J., & Lee, W. (2003). “Sustaining availability of web services under distributed denial of service Attacks”. IEEE Transactions on Computers, 52(2), 195–208.
3. Tamas Abraham. “IDDM: Intrusion Detection Using Data Mining Techniques”. Technical Report DSTO-GD-0286. DSTO Electronics and Surveillance Research Laboratory. 2001.
4. Daniel Barbarra, Julia Couto, Sushil Jajodia, Leonard Popyack, and Ningning Wu (2001). “ADAM: Detecting Intrusions by Data Mining”. Proceedings of the 2001 IEEE, Workshop on Information Assurance and Security, NY, USA.
5. KDD Cup 1999 (2009). Available on: http://kdd.ics.uci.edu/databases/kddcup99/kddcu p99.html.
6. Lan Guo, Yan Ma, Bojan Cukic, and Harshinder Singh. “Robust Prediction of Fault-Proneness by Random Forests”. Proceedings of the 15th International Symposium on Software Reliability Engineering (ISSRE'04). pp. 417-428, Brittany, France.
7. Janhavi Kaskar, Ruchit Bhatt, Rohit Shirsat (2014). “A System for Detection of Distributed Denial of Service (DDoS) Attacks using KDD Cup Data Set”. International Journal of Computer Science and Information Technologies, Vol. 5 (3). Pp. 3551-3555.

8. Mohammad Khubeb Siddiqui and Shams Naahid (2013). "Analysis of KDD CUP 99 Dataset using Clustering based Data Mining". International Journal of Database Theory and Application Vol.6, No.5. pp.23-34.
9. Keunsoo Lee, Juhyun Kim, Ki Hoon Kwon, Younggoo Han, Sehun Kim (2008). "DDoS attack detection Method using Cluster Analysis". Elsevier, Expert Systems with Applications 34. Pp. 1659–1665.
10. Rui Zhong and Guangxue Yue (2010). "DDoS Detection System Based on Data Mining". Proceedings of the Second International Symposium on Networking and Network Security (ISNNS '10) Jinggangshan, P. R. China. pp. 062-065.
11. KDD Cup 1999 (2007). Available on: http://kdd.ics.uci.edu/databases/kddcup 99/kddcup99.html.
12. S. J. Stolfo, W. Fan, W. Lee, A. Prodromidis, and P. K. Chan. "Costbased modeling for fraud and Intrusion Detection: Results from the jam project". discex, vol. 02. pp. 1130, 2000.
13. R. P. Lippmann, D. J. Fried, I. Graf, J. W. Haines, K. R. Kendall, D. McClung, D. Weber, S. E. Webster, D. Wyschogrod, R. K. Cunningham, and M. A. Zissman. "Evaluating Intrusion Detection Systems: The 1998 darpa off-line Intrusion Detection Evaluation". discex, vol. 02. pp. 1012, 2000.
14. Mahbod Tavallaee, Ebrahim Bagheri, Wei Lu, and Ali A. Ghorbani (2009). "A Detailed Analysis of the KDD CUP 99 Data Set". Proceedings of the 2009 IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA).
15. Lewis, P. M. (1962). "The Characteristic Selection Problem in Recognition System". IRE Transaction on Information Theory, 8. Pp. 171-178.
16. John, G.H. et al. (1994). "Irrelevant Features and the Subset Selection Problem". Proc. of the 11th Int. Conf. on Machine Learning, Morgan Kaufmann Publishers. Pp. 121-129.
17. Dash, M. & Liu, H. (1997). "Feature Selection for Classification". Intelligent Data Analysis, 1(3). Pp. 131– 56.
18. Megha Aggarwal, Amrita (2013). "Performance Analysis of Different Feature Selection Methods in Intrusion Detection", International Journal of Scientific & Technology Research volume 2, issue 6.
19. Mihui Kim, Hyunjung Na, Kijoon Chae, Hyochan Bang, Jungchan Na (2004). "A Combined Data Mining Approach for DDoS Attack Detection". H.-K. Kahng and S. Goto (Eds.): ICOIN 2004, LNCS 3090, pp. 943–950.
20. Iba, Wayne and Langley, Pat (1992); Induction of One-Level Decision Trees, in ML92: Proceedings of the Ninth International Conference on Machine Learning, Aberdeen, Scotland, 1–3 July 1992, San Francisco, CA: Morgan Kaufmann, pp. 233–240.

