Machine Learning Based Internet Traffic Recognition with Statistical Approach

Jaiswal Rupesh Chandrakant
Department of Electronics & Telecommunication
Pune Institute of Computer Technology
Pune, India
rcjaiswal@pict.edu

Lokhande Shashikant. D.
Department of Electronics & Telecommunication
Sinhgad College of Engineering
Pune, India
sdlokhande.scoe@sinhgad.edu

***Abstract*— The researchers have started looking for Internet traffic recognition techniques that are independent of 'well known' TCP or UDP port numbers, or interpreting the contents of packet payloads. Newer approaches classify traffic by recognizing statistical patterns in externally observable attributes of the traffic (such as typical packet lengths and inter-arrival times). The main goal is to cluster or classify the Internet traffic flows into groups that have identical statistical properties. The need to deal with Traffic patterns, large datasets and Multi-dimensional spaces of flow and packet attributes is one of the reasons for the introduction of Machine Learning (ML) techniques in this field. ML techniques are subset of Artificial Intelligence used for traffic recognition. Further, there are four types of Machine Learning, i.e. Classification (Supervised learning), clustering (Un-Supervised learning), Numeric prediction and Association. In this research paper IP traffic recognition through classification process is implemented. Different researchers are calling this process as IP traffic Recognition, IP traffic Identification, and sometimes IP traffic classification. Here Real time internet traffic has been captured using packet capturing tool and datasets has been developed. Also few standard datasets have been used in this research work. Then using standard attribute selection algorithms, a reduced statistical feature dataset has been developed. After that, Six ML algorithms AdaboostM1, C4.5, Random Forest tree, MLP, RBF and SVM with Polykernel function classifiers are used for IP traffic classification. This implementation and analysis shows that Tree based algorithms are effective ML techniques for Internet traffic classification with accuracy up to of 99.7616 %.**

Keywords— Internet Traffic Classification, Machine Learning, AdaboostM1, C4.5, Random Forest tree, MLP, RBF and SVM with Polykernel function.

I. INTRODUCTION

Over the past few years a dramatic and rapid increase in the number of internet users, uses various internet applications like VoIP (Google Talk and Skype), Multimedia Streaming (YouTube, Windows media player Streaming, Real player Streaming), bulk data transfer (p2p-torrent, FTP), Interactive traffic (ssh, rlogin, telnet, instant messaging, online Games), Email services (IMAP, POP3, SMTP), WWW traffic (HTTP, HTTPS) and Database traffic (Oracle and DNS). Attackers are creating attack traffic like TCP syn flood, ICMP smurf, and UDP flood traffic. IP Traffic Recognition is an emerging issue for various educational organizations, corporate organizations, government organizations, and internet service providers for various processes such as bandwidth misuse finding, Network congestion, bandwidth planning, Quality of service implementation, Economical solution, effective service pricing mechanisms, network monitoring, fault Diagnosis, Network analysis, Network administration, Lawful inspection[1] and Intrusion detection system [2][3]etc.

Currently there are different approaches for traffic recognition. The most common method is based on recognizing a well known port number from TCP or UDP header of given traffic [4]. The typical port number and classified protocol applications are like, 20-FTP data, 21-FTP control, 22-SSH, 23-Telnet, 25-SMTP, 53-DNS, 80-HTTP, 110-POP3, 113-IRC, 1214-Kazza p2p, 6346-Gnutella p2p traffic. Nowadays, traffic uses dynamically allocated port numbers. For example, hacker can compromise the system & use port 5353 for remote execution instead of standard port 23, and thus Wrong Recognition results. Many Users are hiding their application traffic behind the HTTP web traffic over TCP port 80, in order to get through security products like firewalls & network security tools [5]. That is something else like P2P, VoIP, Internet-TV carried over TCP port 80, and thus Wrong Recognition results using this method. Although port-based traffic recognition is the fastest and simple method, it gives less than 70% accuracy [6][7].

In another method the internet traffic recognition is based on predefined payload signatures to recognize the application services [8][9]. The typical payload signature and classified applications are given as, '\GET'-http, '0x13Bit'-Bit torrent p2p, 'PNG'0x0d0a-MSN messenger, 'USERHOST'-IRC, 'ARTICLE'-nntp and 'SSH'-ssh internet traffic. First limitation with payload signature based methods is that it is difficult to maintain huge database. Also '\GET' signature finds both HTTP and Gnutella traffic, and thus Wrong Recognition results. Also pattern matching with every IP packet consumes the hardware and software resources, makes IP traffic recognition system slower. Furthermore, it does not work with encrypted IP traffic. Also some countries also have laws prohibiting network administrators and operators from reading the payload contents of network packets, due to these difficulties; IP Packet payload signature based method is not used.

978-1-4799-2275-8/13/$31.00 ©2013 IEEE

In statistical based recognition approach it is assumed that internet traffic at network layer (IP layer) has statistical properties which are unique for certain classes of applications and enable users to be distinguish it from each other [10]. The statistical features like min, max, mean and standard deviation of Packet lengths and packet inter-arrival time. These Statistical properties are also known as Features or Attributes or Discriminators. The papers of Traffic recognition & identification through classification or clustering using a Statistical Approach [11]-[25] are the basic source of inspiration. The application of ML techniques includes training and testing process. The features are calculated and then the ML classifier is trained with these features with known traffic classes and creates the classifier model known as Memorization process. This model is then used to classify unknown traffic known as testing or Generalization process. Six ML algorithms are used for IP traffic classification with mentioned datasets and Performance analysis is done.

The rest of this paper is organized as follows: section II outlines some information about related work carried out by various researchers in the field of IP traffic recognition. Section III includes introductory information about six classifiers mentioned. Section IV gives overview of internet traffic dataset used briefly. Implementation and result analysis is given in section V. Section VI includes conclusions and future scope.

II. RELATED WORK

Roughan et al. [26] has used the ML algorithms like Nearest Neighbour, Linear Discriminate Analysis (LDA) And Quadratic Discriminant Analysis (QDA). The recognized applications were Real Media Streaming, Telnet, Kazaa, FTP (data), DNS and HTTPS.

Moore and Zuev [17][18][19][20] have used the ML algorithms like Bayesian Techniques. Total of 248 statistical features were calculated. The recognized traffics were P2P torrent, Database, and Mail Services. This technique gives accuracy in the range of 65% to 95%. Also Authors suggested selecting the minimum features for real time traffic recognition. Further it is stated that Bayesian Neural network improves the accuracy of classification.

Nguyen and Armitage [15] have used the Supervised Naive Bayes algorithm. The recognized applications were Game, Telnet, SSH, HTTP, DNS, NTP, HTTPS, SMTP mail and P2P. Authors suggested not to use TCP/UDP port numbers as features. Reductions in the discriminators were suggested for real time traffic recognition.

Chengjie Gu [27] used the Naive Bayes, SMO, REPTree, Back Propagation, and C4.5 algorithms. The recognized applications were HTTP, POP3, FTP, Streaming, Bit Torrent, eMule, PPlive, eDonkey and Game.

Suchul Lee [25] used the k-nearest neighbor, SVM classifier, C4.5 Decision Tree, Bayesian Network and Neural Network algorithms. The recognized applications were HTTP, DNS SMTP, POP, IMAP ICMP, BGP, HTTPS, FTP, Bootp Quake, Half Life game, Age of Empires game, RIP, NetBIOS, SMB, SNMP, and NTP.

Jun Zhang [28] used the Naive Bayes algorithm. The recognized applications were HTTPS, Bit Torrent, HTTP, SMTP and POP3. This study shows a solution to achieve high-performance IP traffic classification with less training samples labeling. Also few classifiers are taking several hours to train the existing data [29], hence are not recommended for real time classification.

Thus there is scope of further improvement in performance by correct selection and calculation of attributes. Also correct selection and proper implementation of ML algorithms results in improved and accurate internet traffic recognition.

III. MACHINE LEARNING CLASSIFIERS

In this paper, six well-known machine learning algorithms are used which are explained in brief as follows:

A. AdaBoost.M1 classifier

AdaBoost is known as Adaptive Boosting, is a machine learning algorithm. It is a meta-algorithm, and can be used in conjunction with many other learning algorithms to improve their performance. The algorithm is as follows:

1. Start
2. Initialize the observation weights W_i
   weights $W_i = 1/N, i = 1,2,..,N$
3. For $m = 1$ to M, *repeat the following loop*
   {
   Fit a classifier $Z_m(x)$
   To training Discriminator data using weights W_i
   Compute $Err_m = \frac{\sum_{i=1}^{N} W_i \, I(Y_i \neq Z_m(X_i))}{\sum_{i=1}^{N} W_i}$
   Compute $\alpha_m \log\left[\frac{(1 - Err_m)}{Err_m}\right]$
   Set $W_i \leftarrow W_i \; exp\,[\alpha_m \bullet I(Y_i \neq Z_m(X_i))]$
   Where $i = 1,2,\ldots,N.$
   }
4. Output $Z(x) = \text{sign}\,[\sum_{m=1}^{M} \alpha_m Z_m(x)]$
5. End

B. The C 4.5 classifier

C4.5 is an algorithm used to generate a decision tree for classification. C4.5 is an extension of Iterative Dichotomiser- 3 algorithm. The algorithm is as follows:

1. Let classes $C_j = \{C_1, C_2, \ldots, C_k\}$.
2. Input the Training samples T from classes C_j
3. Test – entropy: {If S = set of samples, freq = (C_i, S), $|S|$ = number of samples in the set S.
4. Calculate the entropy of set = Info(S) =
   $\sum ((\text{freq}(C_i, S) / |S|) \cdot \log_2 (\text{freq}(C_i, S)/ |S|))$
5. Test the attribute $\text{Info}_x(T) = \sum ((|Ti| / |T|) \cdot \text{Info}(Ti))$
6. Calculate the $\text{Gain}(X) = \text{Info}(T) - \text{Info}_x(T)$
   (Select an attribute with the highest Gain value).
7. Create decision tree model according to the attributes dataset classification.
8. End.

C. Random Forest classifier

It is an ensemble learning method used for internet traffic classification. It is operated by constructing a decision trees at training time and outputting the mode of class by individual trees. The algorithm is as follows:

1. Start
2. Draw *ntree* bootstrap samples from the original training data.
3. For each of the bootstrap samples, grow an unpruned classification or regression tree.
4. Predict new data by aggregating the predictions of the *ntree* trees (majority votes)
5. Estimate the error rate based on the training data.
6. End

D. MLP classifier

A multilayer perceptron (MLP) is a feed forward ANN model that maps sets of input training attributes onto a set of appropriate outputs class[32][33][34].An MLP consists of different layers of information processing nodes like input, hidden and output with each layer fully connected to the next one. The algorithm is as follows:

1. Start
2. Initialize total error to 0.
3. Apply first pattern as the input and train the neural network.
4. Get error pattern for each output neuron in network and calculate the total error.
5. If last pattern has trained, start again with first pattern otherwise load next pattern and train.
6. If last has trained then,
   {If total error is less than target error then,
   STOP Training.
   Else
   Repeat steps 2-5}
   Else if last pattern has not trained then repeat 5.
7. Then apply the test patterns as the input to neural network to get the classified results.
8. End.

E. RBF classifier

A radial basis function network [32][33][34] is an ANN that uses radial basis functions as activation functions for hidden and output layer neurons. The output of the network is a linear combination of radial basis functions of the inputs and neuron parameters. A radial basis function network training algorithm is as follows:

1. Start
2. Initialize he weights to random values
3. While stopping is false to perform step 4-11
4. For each input perform step 5-10
5. Each input unit (X_i , i=1….n) receives input signals to All Units in the layer above hidden unit.
7. Choose the centers for the RBF functions. Calculate output of I_m unit V_i (X_i) in the hidden layer.
8. Initialize the weights in the output layer
9. Calculate the output of the neural network
10. Calculate error and test the stopping condition
11. End

F. SVM classifier

The support vector machine (SVM) is a classification technique. SVMs with kernels provide a classification with the least amount of error. Thus SVM can work as a Linear Discriminant function as well as Maximum Margin Classifier. SVM classification process algorithm is as follows:

1. Start
2. Set up the training data if input samples
3. Set up SVM's parameters (linearly separable or non-linearly separable data. Use kernel function accordingly)
4. Train the SVM to build the SVM model
5. Classify an input sample using a trained SVM
6. Obtain information about the support vectors using an index and mark them.
7. End.

IV. INTERNET TRAFFIC DATASETS

In this research work, a packet capturing tool, Wire shark, [30] is installed on institute Proxy server to capture entire real time internet traffic. It is configured in non promiscuous mode with "no broadcast and no multicast", filter option selected. In this process of developing datasets, reduced feature dataset is obtained as explained in research papers [17][18][19][20]. Also standard Game datasets are utilized which are already used in papers [15][17][18][19][20]. Further details of the original hand-classification process is given in [17][18][19][20]. CfsSubsetEval attribute evaluator and Best First search method is used in attribute selection process of Weka tool [31].

In this research work, 2.5 GHz Intel Quad CPU workstation with 2GB of RAM and Ubuntu 10.04 (lucid) version operating system is used. GCC compiler and open source OCTAVE tool is also used for this research work. TCP/IP header information is read from "PCAP" datasets using "C" program, Features are calculated in Octave and WEKA compatible file in ARFF format is developed for further research work.

V. IMPLEMENTATION AND RESULT ANALYSIS

A. Methodology

In this research work, Weka tool [31] is used for implementation of IP traffic classification process with six different Machine Learning algorithms and 10 statistical features. The features are Packets/sec, Mean IP packet length, Mean IP payload length, Bytes/Flow, Flow duration, Mean IAT, Bytes/sec, SD of IP packet length, SD of IAT, SD of IP payload length. Total 25558 samples are taken for training-testing purpose and cross validation is done with 10 folds option and classification accuracy is validated. The evaluation Metrics is explained in simple fashion.

Classified as →	A	$\overline{A}$
A	TP	FN
$\overline{A}$	FP	TN

Fig. 1. Performance Evaluation Metrics.

- True positives (TP): Correctly identified instances.
- True negatives (TN): Correctly rejected instances.
- False positives (FP): Incorrectly identified instances.
- False Negatives (FN): Incorrectly rejected instances.

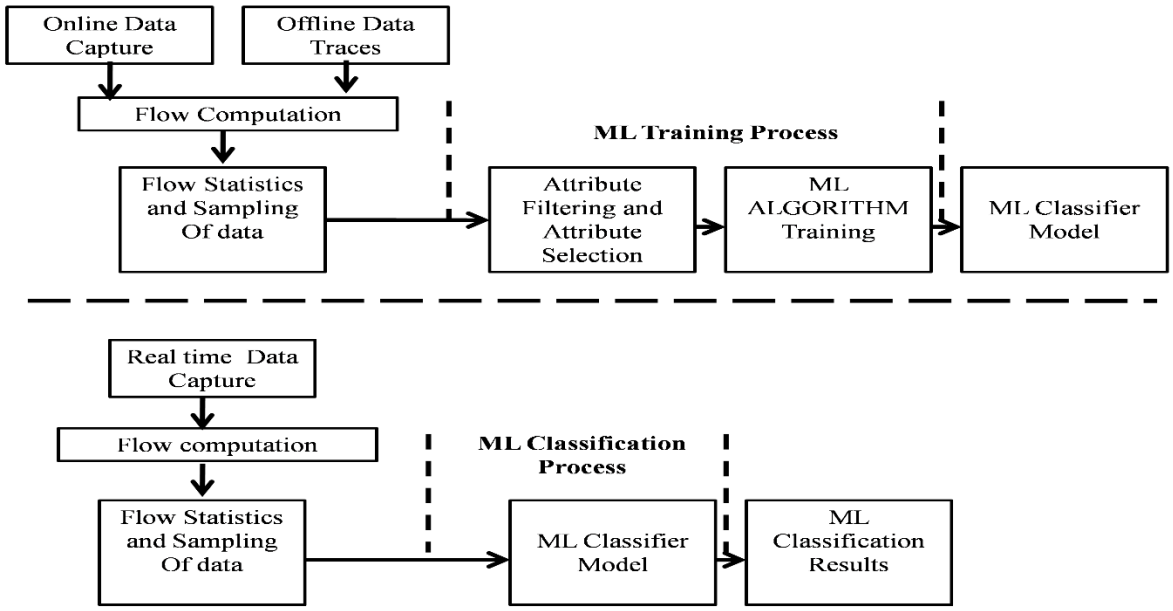

Fig. 1(A). ML Training and ML Classification Process

We captured online internet traffic as shown in Fig. 1(B).

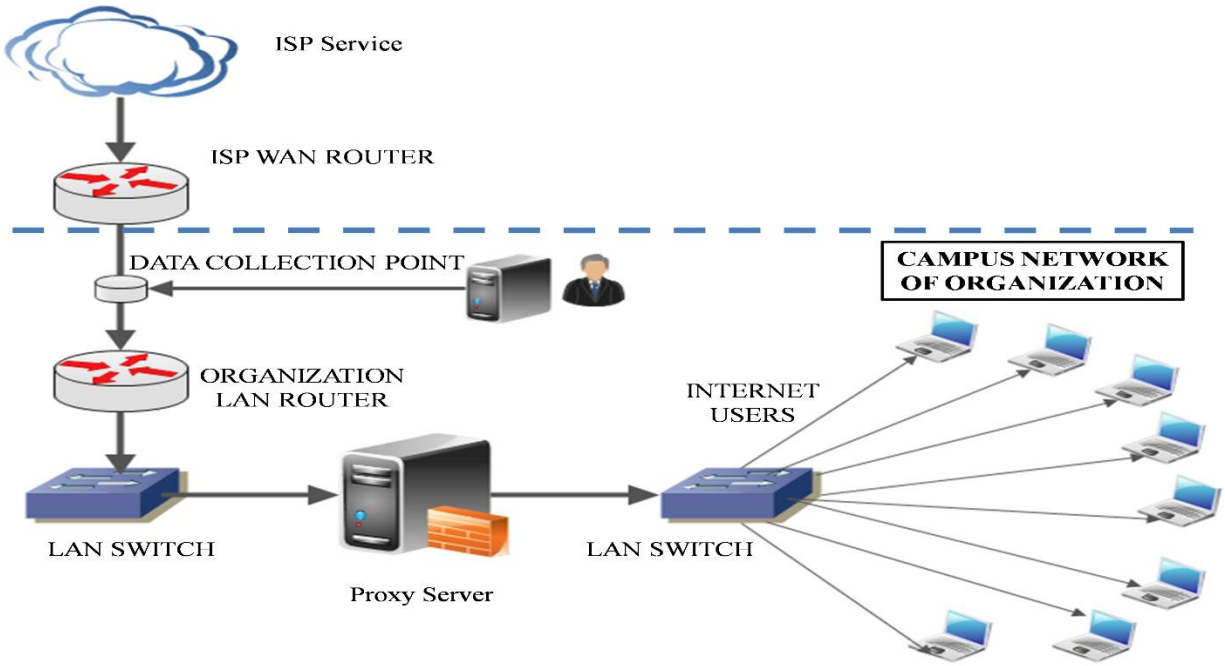

Fig. 1(B). Real Time Traffic capturing Process in Campus

Table 1. Composition of publicly available Data traces.

DATASETS	YEAR	TYPE	VOLUME OF TRAFFIC
MAWI-X	7 OCT.2012	BACKBONE	1350.68 MB
MAWI-Y	7 OCT.2012	BACKBONE	1412.45 MB
LBNL-X	14th OCT. 2005	LAN	370.74 MB
LBNL-Y	14th OCT. 2005	LAN	467.95 MB
CAIDA-X	2008-JULY	BACKBONE	1342.55 MB
CAIDA-Y	2008-JULY	BACKBONE	2117.73 MB
UNIBS-X	30th SEPT.2009	LAN	721.21 MB
UNIBS-Y	30th SEPT. 2009	LAN	786.16 MB

Table 2. Summary statistics of the datasets used for training and testing.

Protocols	Class	Flow%	Training Instances	Testing Instances	Total Instances
BT, edonkey,	p2p	23.7	34078	17556	51634
smtp,pop,imap	mail	4.9	7045	3630	10675
skype,gtalk	VoIP	7.4	10640	5482	16122
dns.whois	dns	15.6	22431	11556	33987
ssl, ssh,	secured	3.3	4744	2445	7189
http, gopher	www	22.9	32935	16960	49895
ftp,xunlei	bulk	6.5	9346	4815	14161
quake,hl,et	games	11.4	16391	8445	24836
rlogin,klogin, tenet,irc	Interactive	4.3	6182	3186	9368

TABLE I
A SUMMARY OF RESEARCH REVIEWED IN SECTION IV

Work	ML Algorithms	Features	Data Traces	Traffic Considered	Classification Level
McGregor et al. [48]	Expectation Maximization	• Packet length statistics (min, max, quartiles, ...) • Inter-arrival statistics • Byte counts • Connection duration • Number of transitions between transaction mode and bulk transfer mode • Idle time Calculated on full flows	NLANR and Waikato trace	A mixture of HTTP, SMTP, FTP (control), NTP, IMAP, DNS ...	Coarse grained (bulk transfer, small transactions, multiple transactions ...)
Zander et al. [46]	AutoClass	• Packet length statistics (mean and variance in forward and backward directions) • Inter-arrival time statistics (mean and variance in forward and backward directions) • Flow size (bytes) • Flow duration Calculated on full-flows	Auckland-VI, NZIX-II and Leipzig-II from NLANR	Half-Life, Napster, AOL, HTTP, DNS, SMTP, Telnet, FTP (data)	Fine grained (8 applications studied)
Roughan et al. [18]	Nearest Neighbour, Linear Discriminate Analysis and Quadratic Discriminant Analysis	• Packet Level • Flow Level • Connection Level • Intra-flow/Connection features • Muli-flow features Calculated on full flows	Waikato trace and section logs from a commercial streaming services	Telnet, FTP (data), Kazaa, Real Media Streaming, DNS, HTTPS	Fine grained (three, four and seven classes of individual applications)
Moore and Zuev [14]	Baysian Techniques (Naive Bayes and Naive Bayes with Kernel Estimation and Fast Correlation-Based Filter method)	Total of 248 features, among them are • Flow duration • TCP port • Packet inter-arrival time statistics • Payload size statistics • Effective bandwidth based upon entropy • Fourier transform of packet inter-arrival time Calculated on full flows	Proprietary Hand Classified Traces	A large range of Database, P2P, Buck, Mail, Services, ... traffic	Coarse grained
Barnaille et al. [53]	Simple K-Means	Packet lengths of the first few packets of bi-directional traffic flows	Proprietary traces	eDonkey, FTP, HTTP, Kazaa, NTP, POP3, SMTP, SSH, HTTPS, POP3S	Fine grained (10 applications studied)
Park et al. [44] [44]	Naive Bayes with Kernel Estimation, Decision Tree J48 and Reduced Error Prunning Tree	• Flow duration • Initial Advertised Window bytes • Number of actual data packets • Number of packets with the option of PUSH • Packet lengths • Advertised window bytes • Packet inter-arrival time • Size of total burst packets	NLANR, USC/ISI, CAIDA	WWW, Telnet, Chat (Messenger), FTP, P2P (Kazaa, Gnutella), Multimedia, SMTP, POP, IMAP, NDS, Oracle, X11	N/A (comparison work)
Nguyen and Armitage [56]	Supervised Naive Bayes	• Packet lengths (min, max, mean, standard deviation) • Inter-Packet lengths statistics (min, max, mean, standard deviation) • Packet Inter-arrival times statistics (min, max, mean, std dev.) • Calculated over a small number (e.g. 25 packets) of consecutive packets (classification windows) taken at various points of the flow lifetime - where the changes in flow's characteristics are significant	Traces collected at an online game server in Australia and provided by University of Twente, Netherland	Online Game (Enemy Territory) traffic, Others (HTTP, HTTPS, DNS, NTP, SMTP, Telnet, SSH, P2P ...)	Application specific (Online Game, UDP based, First Person Shooter, Enemy Territory traffic)

TABLE II
A SUMMARY OF RESEARCH REVIEWED IN SECTION IV(CONTINUED)

Work	ML Algorithms	Features	Data Traces	Traffic Considered	Classification Level
Nguyen and Armitage [54]	Naive Bayes and Decision Tree in combination with Clustering algorithms for automated sub-flows selection	• Packet lengths statistics (min, max, mean, std dev.) • Inter-Packet lengths statistics (min, max, mean, std dev.) • Packet Inter-arrival times statistics (min, max, mean, std dev.) • Calculated over a small number (e.g. 25 packets) of consecutive packets (classification windows) taken at various points of the flow lifetime - where the changes in flow's characteristics are significant. • Further extension with synthetic mirroring features.	Traces collected at an online game server in Australia and provided by University of Twente, Netherland	Online Game (Enemy Territory) traffic, Others (HTTP, HTTPS, DNS, NTP, SMTP, Telnet, SSH, P2P ...)	Application specific (Online Game, UDP based, First Person Shooter, Enemy Territory traffic)
Erman et al. [47]	K-Means	• Total number of packets • Mean packet length • mean payload length excluding headers • Number of bytes transferred • Flow duration • Mean inter-arrival time	Self-collected 8 1-hour campus traces between April 6-9, 2006	Web, P2P, FTP, Others	Coarse grained (29 different protocols grouped into a number of application categories for studies)
Crotti et al. [61]	Protocol fingerprints (Probability Density Function vectors) and Anomaly score (from protocol PDFs to protocol fingerprints)	• Packet lengths • Inter-arrival time • Packet arrival order	6-month self-collected traces at the edge gateway of the University of Brescia data centre network	TCP applications (HTTP, SMTP, POP3, SSH)	Fine grained (four TCP protocols)
Haffner et al. [57]	Naive Bayes, AdaBoost, Regularized Maximum Entropy	Discrete byte encoding of the first n-bytes payload of a TCP unidirectional flow	Proprietary	FTP (control), SMTP, POP3, IMAP, HTTPS, HTTP, SSH	Fine grained
Ma et al. [66]	Unsupervised learning (*product distribution*, *Markov processes*, and *common substring graphs*)	Discrete byte encoding of the first n-bytes payload of a TCP unidirectional flow	Proprietary	FTP (control), SMTP, POP3, IMAP, HTTPS, HTTP, SSH	Fine grained
Auld et al. [55]	Bayesian Neural Network	246 features in total, including: • Flow metrics (duration, packet-count, total bytes) • Packet inter-arrival time statistics • Size of TCP/IP control fields • Total packets in each direction and total for bi-directional flow • Payload size • Effective bandwidth based upon entropy • Top-ten Fourier transform components of packet inter-arrival times for each direction • Numerous TCP-specific values derived from tcptrace (e.g. total payload bytes transmitted, total number of PUSHED packets, total number of ACK packets carrying SACK information etc.)	Proprietary hand classified traces	A large range of Database, P2P, Buck, Mail, Services, Multimedia, Web ... traffic	Coarse grained

TABLE III
A SUMMARY OF RESEARCH REVIEWED IN SECTION IV(CONTINUED)

Work	ML Algorithms	Features	Data Traces	Traffic Considered	Classification Level
Williams et al. [65]	Naive Bayes with Discretisation, Naive Bayes with Kernel Estimation, C4.5 Decision Tree, Bayesian Network and Naive Bayes Tree	• Protocol • Flow duration • Flow volume in bytes and packets • Packet length (minimum, mean, maximum and standard deviation) • Inter-arrival time between packets (minimum, mean, maximum and standard deviation)	NLANR	FTP(data), Telnet, SMTP, DNS, HTTP	N/A (Comparison work)
Erman et al. [45]	K-Means, DBSCAN and AutoClass	• Total number of packets • Mean packet length • Mean payload length excluding headers • Number of bytes transfered (in each direction and combined) • Mean packet inter-arrival time	NLANR and a self-collected 1-hour trace from the University of Calgary	HTTP, P2P, SMTP, IMAP, POP3, MSSQL, Other	N/A (Comparison work)
Erman et al. [64]	Naive Bayes and AutoClass	• Total number of packets • Mean packet length (in each direction and combined) • Flow duration • Mean data packet length • Mean packet inter-arrival time	NLANR	HTTP, SMTP, DNS, SOCKS, FTP(control), FTP (data), POP3, Limewire	N/A (Comparison work)
Bonfiglio et al. [67]	Naive Bayes and Pearson's Chi-Squate test	• Message size (the length of the message encapsulated into the transport layer protocol segment) • Average inter packet gap	Two self collected datasets	Skype traffic	Application specific

TABLE IV
REVIEWED WORK IN LIGHT OF CONSIDERATIONS FOR OPERATIONAL TRAFFIC CLASSIFICATION

Work	Real-time Classification	Feature Computation Overhead	Classify Flows In Progress	Directional neutrality
McGregor et al. [48]	No	Average	Not addressed	No
Zander et al. [46]	No	Average	Not addressed	No
Roughan et al. [18]	No	Average	Not addressed	N/A
Moore and Zuev [14]	No	High	Not addressed	No
Barnaille et al. [53]	Yes	Low	Not addressed	No
Park et al. [44]	No	Average	Not addressed	Not clear
Nguyen and Armitage [56]	Yes	Average	Yes	Yes
Nguyen and Armitage [54]	Yes	Average	Yes	Yes
Erman et al. [47]	No	Average	Not addressed	No
Crotti et al. [61]	Yes	Average	Not addressed	No
Haffner et al. [57]	Yes	Average	Not addressed	N/A
Ma et al. [66]	No	Average	Not addressed	No
Auld et al. [55]	No	High	Not addressed	No
Williams et al. [65]	N/A	Average	N/A	N/A
Erman et al. [45]	N/A	Average	N/A	N/A
Erman et al. [64]	N/A	Average	N/A	N/A
Bonfiglio et al. [67]	Yes	Average	Not addressed	Not clear

- Thus recall is the ratio of relevant instances retrieved to total number of relevant instances available. Whereas Precision is the ratio of relevant instances retrieved to total number of relevant and irrelevant instances available.
- Recall is given as : $\text{Recall} = \frac{\text{TP}}{\text{TP+FP}}$
- Precision is given as :$\text{Precision} = \frac{\text{TP}}{\text{TP+FN}}$
- Accuracy is given as :
  $$\text{Acc.} = \frac{\text{Number of correct prediction of instances}}{\text{Total number of instances considered}}$$
- F-measure is given as :
  $$\text{F} - \text{Measure} = \frac{(\beta^2+1)*\text{P}*\text{TP}}{(\beta^2*\text{P})+\text{TP}}$$
  β can have value 0 to ∞, and it is used to control weight assigned to P and TP.
- ROC is known as *Receiver operating characteristic* for a certain class label of dataset and plotted as ROC Graphs. It is a plot with the FP rate on the *X* axis and the TP rate on the *Y* axis.
- Mean absolute error (MAE) is used to measure how close forecasts (or predictions) are to the eventual (or true) value of outcomes and given as :
  $$\text{MAE} = \frac{1}{\text{n}} \sum_{\text{i}=1}^{\text{n}} |(\text{P}_\text{i} - \text{E}_\text{i})|$$
  Where P_i is the predicted value of instance and E_i is the true value of instance.
- Mean square error (MSE) measures the average of squares of the errors. is given as :
  $$\text{MSE} = \frac{1}{\text{n}} \sum_{\text{i}=1}^{\text{n}} (\text{P}_\text{i} - \text{E}_\text{i})^2$$
  Where P_i is the predicted value of instance and E_i is the true value of instance.
- Root Mean square error (RMSE) is the root of MSE and is given as :
  $$\text{RMSE} = \sqrt{\frac{1}{\text{n}} \sum_{\text{i}=1}^{\text{n}} (\text{P}_\text{i} - \text{E}_\text{i})^2}$$
  Where P_i is the predicted value of instance and E_i is the true value of instance

B. Results and Analysis

It is clear from this figure 2 that maximum accuracy is provided by Random Forest tree classifier which is 99.7616 %. The C4.5 classifier gives 98.468 % and 98.4641 % accuracy for full and reduced feature dataset respectively. Figure 3 and Figure 4 indicates comparison of training time required for six and five ML classifiers respectively. It is evident that training time of AdaboostM1 classifier is least. The highest training time is taken by RBF classifier. Also it is evident that training time of all classifiers is decreased for reduced feature dataset compared to full feature dataset. Figure 5 indicates that Mean Absolute Error (MAE) of Random Forest tree classifier is least and maximum for SVM classifier. Figure 6 indicates that the Root Mean Squared Error (RMSE) of Random Forest tree classifier is least and maximum for SVM classifier. Figure 7 and Figure 8 indicates comparison of Weighted Precision, Recall, F-measure and ROC values for six ML classifiers for full and reduced feature dataset respectively. It is evident that maximum values are resulted for Random Forest tree and C4.5 classifier. Least values are observed for AdaboostM1 and SVM classifiers. It is also observed that, these values of all classifiers are almost same for full and reduced feature dataset. Figure 9 indicates that the maximum Precision is resulted for attack traffic and minimum for Skype traffic. It is evident that precision value for full feature datasets are more compared to reduced feature datasets except for streaming traffic. Figure 10 indicates that the maximum Recall value is resulted for attack and DNS traffic and minimum for Skype traffic. Figure 11 indicates that the maximum F-measure value is resulted for attack, DNS and Gaming and minimum for Skype traffic.

Figure 12 indicates that the maximum ROC value is resulted for attack and Gtalk traffic and minimum for Skype traffic and p2p torrent traffic. For Telnet traffic the ROC value is more for full feature dataset than reduced feature dataset. Figure 13 indicates that the maximum Precision is resulted for attack, IDM and Telnet traffic and minimum for Gaming traffic. It is evident that precision value for gaming traffic for full feature datasets is more compared to reduced feature dataset. It is also observed the reverse case that precision value for Skype traffic for full feature datasets is lesser than reduced feature dataset. Figure 14 indicates that maximum Recall value is resulted for attack, DNS, IDM and Telnet traffic and minimum for p2p torrent traffic. It is also evident that recall values are same for both full feature dataset and reduced feature dataset. Figure 15 indicates that the maximum F-measure value is resulted for attack, DNS, IDM, streaming, Telnet traffic and minimum for Gaming traffic. It is evident that F-measure value for Gaming traffic full feature datasets is more compared to reduced feature dataset. It is also observed the reverse case that F-measure value for Skype traffic for full feature datasets is lesser than reduced feature dataset. Figure 16 indicates that the maximum and same ROC value is resulted for all traffics for full feature dataset and reduced feature dataset which is unique case in this research.

VI. CONCLUSIONS AND FUTURE SCOPE

From these results, it is evident that Random forest tree and C4.5 tree classifiers gives better performance in terms of classification accuracy, training time, Mean Absolute Error, Root Mean Square Error, Precision, Recall, F-measure and ROC area values as compared to other classifiers for both full and reduced feature dataset and can be used for real time classification.

It is evident that the AdaboostM1 is fastest among all these six classifiers but it is not accurate as others. Random Forest Classifier gives 100% ROC value for all nine classes of traffics for both full and reduced feature dataset. Also Random Forest Classifier gives 99.7616 % Accuracy for both full and reduced feature dataset for all nine classes of traffics. Although MLP ML classifier gives 89.9992 % and 88.6783 % accuracy and training time of 171.91 seconds and 127.45 seconds for both full and reduced feature dataset respectively, this performance is not remarkable. There is still scope of further improvement in accuracy and reduction in training time to great extent. These are very important aspects of the real time internet traffic classification process.

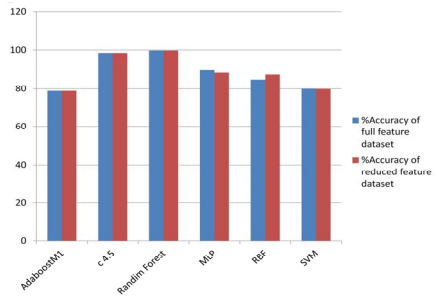

Fig. 2. Comparison of % Accuracy.

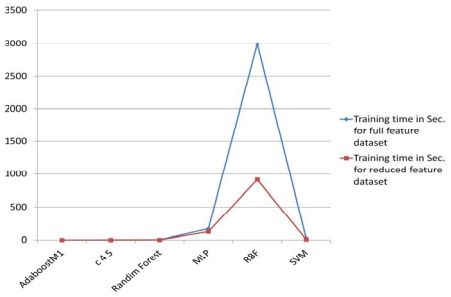

Fig. 3. Comparison of training time.

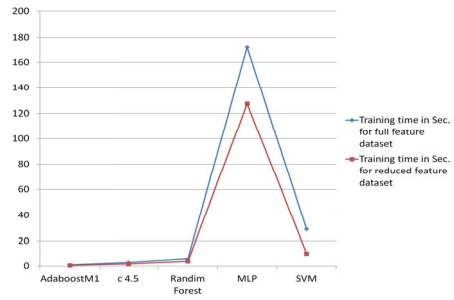

Fig. 4. Comparison of training time w/o RBF.

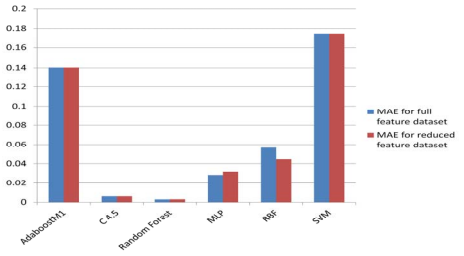

Fig. 5. Comparison of Mean Absolute Error

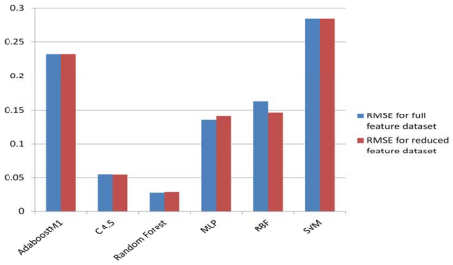

Fig. 6. Comparison of Root Mean Square Error.

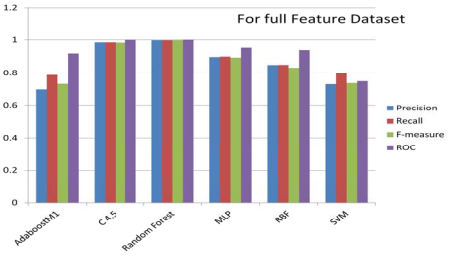

Fig. 7. Weighted factor values for full features.

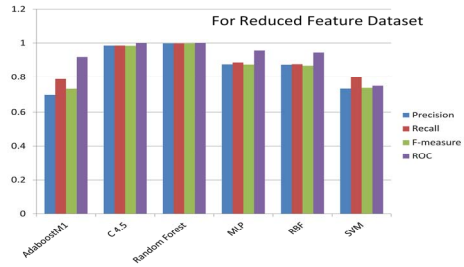

Fig. 8. Weighted factor values for reduced feature.

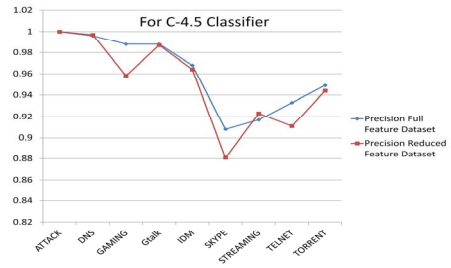

Fig. 9. Precision values for C4.5 classifier.

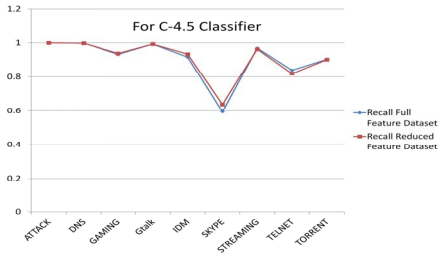

Fig. 10. Recall values for C4.5 classifier.

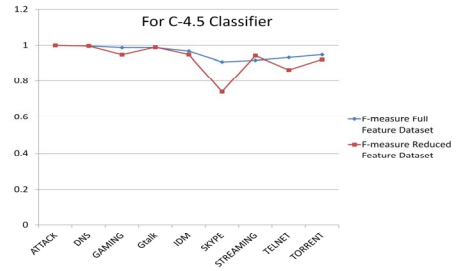

Fig. 11. F-measure values for C4.5 classifier

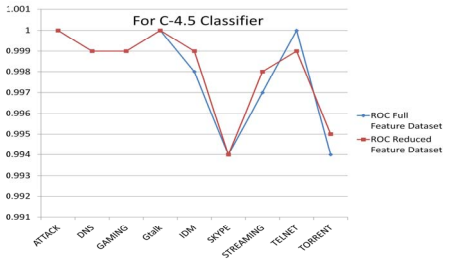

Fig. 12. ROC values for C4.5 classifier.

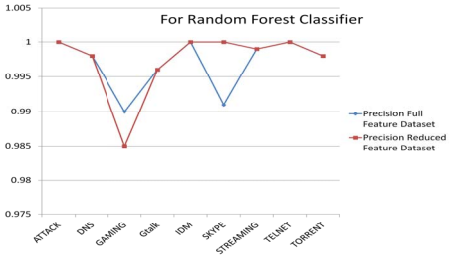

Fig. 13. Precision values of Random Forest.

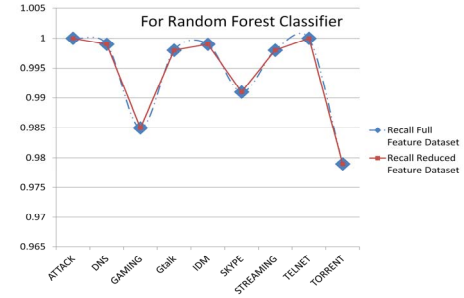

Fig. 14. Recall values of Random Forest.

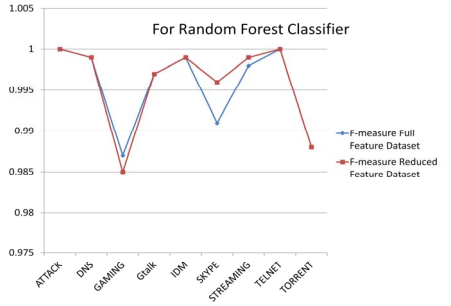

Fig. 15. F-measure values of Random Forest.

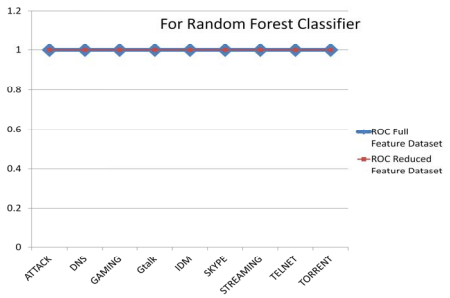

Fig. 16. ROC values of Random Forest.

Also computational complexity can be decreased significantly if no. of attributes used to recognize individual internet application is wisely selected. Same thing is applicable for SVM classifier. Thus performance analysis of these six ML classifiers says that reduced feature dataset reduces the training time significantly and speed up the recognition process to a great extent which requires in real time classification of Internet traffic.

In this research work, internet traffic dataset has been developed by considering packet flow duration of several minutes for individual internet application which is still very large. This duration can be reduced further for fewer packets. Secondly, for all internet traffic applications, the standard datasets were not available. Internet traffic was captured on proxy server in college campus only. Hence IP traffic can also be captured from various real time environments such as laboratories, Staff cabins, Internet server room machines, office, university campus, home environments etc.

The work can be extended for many other internet applications like FTP, E-MAIL, ICMP, IGMP, ARP, RARP, SSH, IMAP, POP-3, SMTP, X-server, TFTP, BOOTP, SNMP, Database transaction, RSVP, RTP, RTCP, RTSP and Trace-route etc.

REFERENCES

[1] F. Baker, B. Foster, and C. Sharp, "Cisco architecture for lawful intercept in IP networks," Internet Engineering Task Force, RFC 3924, 2004.

[2] V. Paxson, "Bro: A system for detecting network intruders in real-time," Computer Networks, no. 31(23-24), pp. 2435–2463, 1999.

[3] Snort - The de facto standard for intrusion detection/prevention, http://www.snort.org, as of August 14, 2007.

[4] IANA.TCP and UDP port numbers, http://www.iana.org/assignments/port-numbers.

[5] Alberto Dainotti, Ludmila I. Kuncheva, Antonio Pescap`e, Carlo Sansone, "Identification of traffic flows hiding behind TCP port 80", IEEE ICC 2010 Proceedings.

[6] H. Dreger, A. Feldmann, M. Mai, V. Paxson, and R. R. Sommer, "Dynamic application layer protocol analysis for network intrusion detection", In USENIX Security Symposium, July 2006.

[7] A. Moore and K. Papagiannaki, "Toward the accurate identification of network applications", In PAM, April 2005.

[8] S. Sen, O. Spatscheck, and D. Wang, "Accurate, scalable in network identification of P2P traffic using application signatures," in WWW2004, New York, NY, USA, May 2004.

[9] Application specific bit strings, http://www.cs.ucr.edu/tkarag/papers/strings.txt.

[10] V. Paxson, "Empirically derived analytic models of wide-area TCP connections", IEEE/ACM Trans. Networking, vol. 2, no. 4, pp. 316–36, 1994.

[11] C. Dewes, A. Wichmann, and A. Feldmann, "An analysis of Internet chat systems," in ACM/SIGCOMM Internet Measurement Conference 2003, Miami, Florida, USA, October 2003.

[12] K. Claffy, "Internet traffic characterisation," PhD Thesis, University of California, San Diego, 1994.

[13] T. Lang, G. Armitage, P. Branch, and H.-Y. Choo, "A synthetic traffic model for half life," in Proc. Australian Telecommunications Networks and Applications Conference 2003- ATNAC2003, Melbourne, Australia, December 2003.

[14] T. Lang, P. Branch, and G. Armitage, "A synthetic traffic model for Quake 3," in Proc. ACM SIGCHI International Conference on Advances in computer entertainment technology (ACE2004), Singapore, June 2004.

[15] Thuy T.T. Nguyen and Grenville Armitage, "A Survey of Techniques for Internet Traffic classification using Machine Learning", IEEE communications surveys & tutorials, vol. 10, no. 4, fourth quarter 2008.

[16] Arthur Callado, Carlos Kamienski Member, IEEE, Géza Szabó, Balázs Péter Ger˝o, Judith Kelner, Stênio Fernandes Member, IEEE, and Djamel Sadok, Senior Member, IEEE. "A Survey on Internet Traffic Identification", IEEE communications surveys & tutorials,vol. 11, no. 3, third quarter 2009.

[17] A. W. Moore and D. Zuev, Discriminators for use in flow-based classification (2005), Intel Research Tech. Rep.

[18] A. Moore and D. Zuev, "Internet traffic classification using Bayesian analysis techniques," in ACM International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS) 2005, Banff, Alberta, Canada, June 2005.

[19] D. Zuev and A. W. Moore, "Traffic classification using a statistical approach," in Proc. 6th Passive Active Meas. Workshop (PAM), Mar. 2005, vol. 3431, pp. 321–324.

[20] T. Auld, A. W. Moore, and S. F. Gull, "Bayesian neural networks for Internet traffic classification," IEEE Trans. Neural Networks, no. 1, pp. 223–239, January 2007.

[21] H. Kim, K. Claffy, M. Fomenkov, D. Barman, M. Faloutsos, and K. Lee, "Internet traffic classification demystified: Myths, caveats, and the best practices," in Proc. ACM CONEXT, Madrid, Spain, Dec. 2008, Article no. 11.

[22] G. Szabó, I. Szabó, and D. Orincsay, "Accurate Traffic Classification," IEEE WOWMOM 2007, Helsinki, Finland, June 18–21, 2007, pp.1–8.

[23] Zhaohong Lai, Alex Galis, Miguel Rio and Chris Todd, "Towards Automatic Traffic Classification", Third International Conference on Networking and Services(ICNS'07), 2007.

[24] J. Hurley E. Garcia-Palacios S. Sezer, "Classifying network protocols: a 'two-way' flow approach", IET Communication, 2011, Vol. 5, Issue. 1, pp. 79–89.

[25] Suchul Lee, Hyun-chul Kim, Dhiman Barman, "NeTraMark: A Network Traffic Classification Benchmark", 2011, ACM SIGCOMM Computer Communication Review.

[26] M. Roughan, S. Sen, O. Spatscheck, and N. Duffield, "Class-of-service mapping for QoS: A statistical signature-based approach to IP traffic classification," in Proc. ACM/SIGCOMM Internet Measurement Conference (IMC) 2004, Taormina, Sicily, Italy, October 2004.

[27] Chengjie Gu, Shunyi Zhuang, Yanfei Sun, Junrong Yan, "Multi-levels Traffic Classification Technique" , 2010 IEEE.

[28] Jun Zhang, Chao Chen, Yang Xiang, "Internet Traffic Classification By Aggregating Correlated Naive Bayes Predictions", IEEE Transactions On Information Forensics And Security, Vol. 8, No. 1, January 2013.

[29] Kuldeep Singh and Sunil Agrawal, "Internet Traffic Classification using RBF Neural Network," in International Conference on Communication and Computing technologies(ICCCT-2011), Jalandhar, India, February 25-26, 2011, paper 10, p.39-43.

[30] Wireshark, Available: http:// www.wireshark.org

[31] Weka website. Available: http:// www.cs.waikato.ac.nz/ml/weka/

[32] S.N.Sivanandam, S.Sumathi and S.N.Deepa, 'Inroduction to neural networks using Matlab 6.0', Tata McGraw Hill Education private Limited, New Delhi, 2009.

[33] Simon Hakin, Neural Networks: A Comprehensive foundation, 2th edition, Pearson Prentice Hall, New Delhi, 2005.

[34] Jang, Sun and Mizutani, Neurofuzzy and soft computing: 1st edition, Prentice Hall, New Delhi, 2012.