Implementation of DBSCAN Algorithm for Internet Traffic Classification

Shezad Shaikh[1], Ashphak P. Khan[2], Vinod S. Mahajan[3]
North Maharashtra University, Jalgaon.
D.N.Patel College of Engineering, Shahada, India.

Abstract: **Traffic modelling and classification find importance in many areas such as bandwidth management, traffic analysis, prediction and engineering, network planning, Quality of Service provisioning and anomalous traffic detection. Much of research work has been done in the area of network traffic classification by application type and several classifiers are suggested. In past network traffic classification using traditional techniques such as well known port number based and payload analysis based techniques are no more effective because various applications uses port hopping and encryption technique to avoid detection. Recently machine learning techniques such as supervised, unsupervised and semi supervised techniques are used to overcome the problems of traditional techniques. In this work we use semi supervised machine learning approach to classify the network traffic using DBSCAN algorithm. This techniques uses only flow statistics to classify the network traffic. This methodology is based on machine learning principle, consists of two components: clustering and classification. The goal of clustering is to partitions the training data set in to two disjoint group flow and traffic class. After making clusters classification is performed in which labelled data are used for assigning class label to the cluster? A NSL-KDD data set is used for testing this approach. Which includes many kinds of attacks and normal data? Experimental result shows that DBSCAN has better effectiveness and efficiency.**

Keyword: **Clustering, Classification, Machine learning, DBSCAN, Traffic classification.**

I. INTRODUCTION

Network traffic classification is the process of identifying traffic flows and associating them to different categories of network application, and it represents an essential task in the whole chain of network management [2]. The aim of network traffic classification is to find out what type of application are run by end users, and what is the share of the traffic generated by different applications in the total traffic mix. All activities related to network are linked to traffic. Network traffic is an important carrier to record and reflect the internet and end user activities; it is also an important composition of network behaviour, through the analysis of network traffic statistics, we can master the network statistical behaviour indirectly. Network traffic classification plays an important role in network activities such as network management, planning and network design. It also includes the allocation, control and management of resources in TCP/IP networks. Network classification is also essential for bandwidth management, traffic shaping, intrusion detection and abnormality. The above activities need the capability of accurately classifying and identifying internet traffic. At the same time accuracy of traffic classifier is a main basis of network security and traffic engineering [1].

In past classical methods such as port number based techniques and payload analysis techniques are more popular to classify the network traffic. The classical well known port number based method [8, 12] is the simplest one. This method needs to access the header of the packet to inspect the port number and recognize the application according to the IANA’s (Internet Assigned Number Authority) this technique fails to classify the traffic accurately because many application uses dynamic port negotiation and because of ambiguity in port number assignment to application by IANA. Payload Analysis [10] technique was introduced to solve the problems of port number based technique. It needs to access the payload of the packet to find the specific pattern in the payload to classify the traffic. This technique fails because various applications use encryption techniques to avoid detection and legality and privacy law does not allow scanning users payload.

Machine learning approach is now uses to classify network traffic. It uses only flow statistics such as duration, protocol types, services, flags etc to classify the traffic and does not need to access the header and payload of the packet. Machine learning approach is classified as [8, 9] unsupervised, supervised and semi supervised approach. Supervised approach needs the labelled instances to train the classifier. Decision tree, Support Vector Machine, Naïve Bayes etc. are supervised algorithms. Supervised approach has following limitations; first, labelled instances are rare and difficult to obtain. Second, it forces mapping of instances to one of the known class without detecting new ones. Unsupervised approach is a class of machine learning in which unlabeled instances are used and based on the inner similarity between instances clusters (groups) are formed. K-Means, DBSCAN, CLARANS etc are unsupervised algorithms. It has limitation in assigning label to cluster after clustering so that new instances properly mapped to applications. Semi supervised approach is a combination of supervised and unsupervised approach [10]. The proposed technique permits both labelled and unlabeled instances to build the traffic classifier.

The rest of this paper is organized is written as followed. Section II represents back ground and related work about internet traffic classification. Section III introduces DBSCAN algorithm. Section IV and V presents our methodology data set and experimental results. Section VI represents our conclusion.

II. BACK GROUND AND RELATED WORK

A. Back ground

Much research work has been done in the field of traffic classification. This section introduces the work related to semi supervised approach. The traditional technique for the traffic classification is based on the well-known port number. Literatures [2-4] have presented that this technique is no longer effective for some applications due to the use of dynamic port number and the camouflage of application. So the port-based approach is currently combined with other approaches for traffic classification. Another familiar technique used widely is based on analysis of packet payloads [2][5-12]. It analyses packet payloads to determine whether they contain the given characteristic strings. The most famous approach that uses this technique is application signature approach, as some known applications contain the specific characteristic signatures [9]. H.Bleul [10-11] shows that measurement systems based on application signatures can provide high accuracy. P.Haffner [6] presents a statistical machine learning algorithm to automatically extract application signatures from IP traffic payload. Although packet payload analysis has high accuracy, it has some shortcomings: (1) it can’t deal with encrypted application payloads; (2) inspecting concerns of packet payloads may be faced with the problem on legality and privacy; (3) it required increased processing and storage capacity; (4) it is useless if payloads are not available; (5) it is unable to identify the unknown application;(6) signatures must be obtained in advance, and it may be changed along with the evolvement of applications. Another technique based on transport layer information of packets is often used, too. This technique usually makes use of connection patterns of the traffic flows, or the role distinction of communication hosts, or the characteristic parameters of network and etc.

T.Karagiannis in [2][5] presents a systematic approach based on the connection pattern of P2P flows and the role distinction derived from the transport layer information. His results show that this approach has the accuracy comparable with that of payload based approach, and is capable of identifying traffic flows missed by payload analysis. It’s main merits are that it has no access to packet payload, and need no knowledge of port numbers and no additional information other than what current flow collectors provide.

F.Constantinou in [11] resents a novel approach for P2P traffic identification that uses fundamental characteristic of P2P protocols, such as a large network diameter and the presence of many hosts acting both as servers and clients. At present, a technique based on machine learning [3] [13-18] has attracted more and more attention. This technique generally consists of two parts: model building and classification. Firstly a model is built by some methods such as clustering, and the clusters are labelled. Secondly a classifier utilizes this model to classify flow data. According to whether the train data classified or not in advance, the machine learning technique may primarily be divided into two types: unsupervised and supervised. When building the model, the supervised approach should classify the train data in advance, but the unsupervised approach not. The machine learning approach is not panacea, we need apply other techniques (e.g. payload analysis, based-on transport layer approaches, and etc.) to label the flows, however.

A.McGregor in [12] presents a machine learning approach, which uses the Expectation Maximization EM clustering algorithm. J.Erman compares three clustering algorithms: K-means, DBSCAN and Auto Class in [3], and applies an supervised approach using EM and Naïve Bayesian classifier in [13]. In all those mentioned, the clustering is one of the most important tools for model-building.

B. *Related work*

This section introduces the work related with semi supervised approach.

In 2007 Jeffery Erman et al.[1,3] proposed a semi supervised traffic classification technique consists of two steps clustering and classification. For experimental purposes traces are collected from the internet link of a large university in which 29 application are identified. The authors categorized traces as 1hour campus, 10 hour residential and 1 hour wireless LAN. They performed various experiment on this work. In first experiment 64000 unlabeled flows are provided for clustering after these flows is clustered, the fix numbers of random flows in each cluster are labelled. The results show that 94% accuracy is achieved by two labelled flows per cluster and K=400. The second set of experiments 80,800 and 8000 labelled flows are mixed with random number of unlabeled flows to generate the training dataset. The accuracy will increase when five or more flows are labelled per cluster.

In 2008 Chuanliang chan et al.[14] proposed two graph based semi supervised methods(i.e. spectral graph transducer, Gaussian fields approach) and one semi supervised clustering (MPCK Means) method to perform intrusion detection .KDD CUP 99 dataset is used for experimental purpose and Precision and Recall and F Measure is used to evaluate the clustering results. The authors compared two semi supervised classification with other traditional supervised algorithm and finds that performance of their approach are much better than other. Also show that the performance of MPCK means is better than K-Means.

In 2009 Levi Lesis and Jorg Sander [18] proposed semi supervised algorithm called as SSDBSCAN. This algorithm requires only one input parameter, does not need user intervention and automatically finds noise objects. The authors used both artificial and real world datasets for experimental purpose and compare SSDBSCAN with HISSCLU and finds that their approach is better to find the cluster in datasets. The Liu bin and Tu Hao in 2010 [19] proposed semi supervised clustering methods based on particle swarm optimization (PSO) algorithm and two host feature name IP address discreteness and success rate of connections. They collected experimental dataset from the router of their university which contained 7 classes. To evaluate their approach they used precision. Result showed that 85% accuracy achieved when 100 or more labeled samples are used in training dataset.

In 2010 Amita Shrivastav and Aruna Tiwari [15] proposed a semi supervised approach based on clustering algorithm. This approach has two steps clustering and classification. KDD CUP-99 dataset is used for experimentation. They compare their approach with SVM based classifier. The experimental result showed that accuracy of proposed classifier lies between 70% and 96% for various datasets.

In 2012 Vinod Mahajan and Bhupendra Verma [6] proposed semi supervised distance based clustering and probabilistic assignment technique for network traffic classification. It permits both labeled and unlabeled instances to be used in training the classifier. The classifier achieves 94.8% accuracy.

C. System Architecture

Before applying clustering, we need to follow few prerequisites. Real word data tend to be dirty, incomplete and inconsistent. Data preprocessing technique can improve the quality of the data, thereby helping to improve the accuracy and efficiency of the subsequent mining process.

Data set can also be in varying from, e.g. one attribute varies in range of 100 and other attribute varies in the range of 10000. So, a proper normalization of data set is done in which each attribute come in the range of 100 or whatever user selects. This normalization technique is known as Min-max normalization.

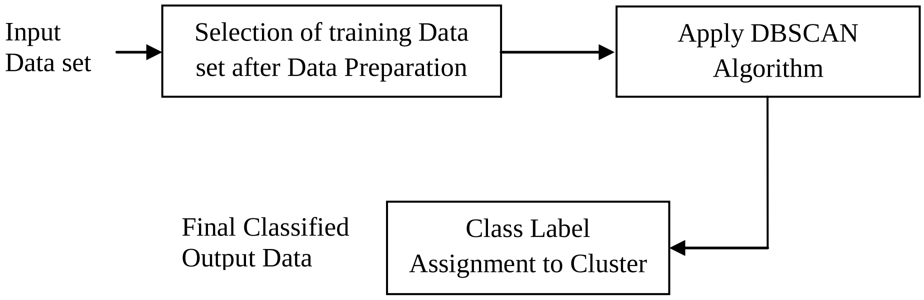
Input
Data set
Selection of training Data set after Data Preparation
Apply DBSCAN Algorithm
Class Label Assignment to Cluster
Final Classified Output Data

Figure.1: Architecture of Proposed System

The input data set is the real data which captured in the real network. It includes many kinds of attack data, also includes the normal data. The classification model is built is based on semi-supervised machine learning approach, thus both labelled and unlabelled data record are present. The output will basically would be the classification that will specify the class to which the data set is belong irrespective of the data input is labelled or unlabelled.

The figure 1. shows the sequential execution of various phases the output of the first phase act as input to the second phase and so on. Initially the input is taken as the data set which acts as input to phase one in which both labelled and unlabelled data are used. It will partition a whole data space into small number of disjoint regions (cluster). Finally, it labels the cluster, for each cluster formed; perform the probabilistic Maintaining the Integrity of the Specifications assignment to find the mapping from cluster to labels. If the maximum priority belongs to same class, then all the labelled and unlabelled data samples within the cluster are assigned the same class label. The result parameter after each phase can be viewed by the user. These results are displayed in the form of graphical tables. The entire classification task terminate once the class are assigned to all the data samples. This classification technique helps in classifying data and also making the system to learn how to classify a new coming data.

III. PROPOSED WORK

In this paper we proposed semi-supervised technique using DBSCAN algorithm which classifies network flows by using only flow statistics which is analysed and implemented. This technique is based on machine learning principle consists of two components clustering and classification. Clustering is used to partitions the training data set into disjoint group (cluster) .After making cluster, classification is performed in which labelled data are used for assigning class labels to the clusters. Labelled data means the input set for which the class to which it belong is known. Unlabelled data set is one for which class to which it belongs is unknown and is to be properly classified. This technique will enable to built a traffic classifier using flow statistic from both labelled and un labelled flow. Our method consists of two step clustering and classification. The details of these steps are as follows.

A. Clustering

We first employ a clustering algorithm to partition a training data set that consist of labelled flows combined with unlabelled flows. Clustering data is method by which the large sets of data are grouped into clusters of small sets of similar data. We propose DBSCAN algorithm for clustering purpose.

B. Classification

After clustering of training data, we use the available labelled flows to obtain a mapping from the clusters to the different known classes the result of the learning is a set of cluster. Some mapped to the different flow types. This method referred to as semi-supervised learning. The input data for classification task is collection of numbers of records. Each record, also known as instance, is characterized by a tuple (x, y) where x is the attribute set and y is class attribute.

IV. DBSCAN ALGORITHM

Density-Based Spatial Clustering and Application with Noise (DBSCAN) was a clustering algorithm based on density. It did clustering through growing high density area, and it can find any shape of clustering. There are two important objects clusters and noise, for DBSCAN algorithm. All points in data set are divided into points of clusters and noise. The key idea of DBSCAN is that for each point of a cluster the neighbourhood of a given radius has to contain at least a minimum number of points, i.e. the density in the neighbourhood has to exceed some threshold. The shape of a neighbourhood is determined by the distance function for two points p and q, denoted by dist(p,q) . We use Euclidean distance function for DBSCAN in this paper as formula.

$$dist(p,q) = \sqrt{\sum_{i=1}^{n}(p_i - q_i)^2}$$

Where n is the number of the features for point object p and q, pi and qi are the ith feature of point object p and q.

A. *Primitives of DBSAN Algorithm*

1. ε-neighbor: the neighbour's in ε semi diameter of an object.
2. Kernel object: certain number (MinP) of neighbour's in ε semi diameter.
3. To a object set D, if object p is the ε-neighbor of q, and q is kernel object, then p can get "direct density reachable" from q.
4. To a ε, p can get "direct density reachable" from q; D contains Minp objects; if a series object p1,p2,..,pn , p1= q, p n =p, then Pi+1 can get "direct density reachable" from pi,, pi €i D, 1 ≤ i ≤ n.
5. To ε and MinP, if there exist a object o(o€D) , p and q can get "direct density reachable" from o, p and q are density connected.

The DBSCAN algorithm is based on the concepts of density reach ability and density-connectivity. These concepts depend on two input parameters: epsilon (eps) and minimum number of points (minpts). Epsilion is the distance around an object that defines its eps- neighbourhood. For a given object q, when the number of objects with in the eps-neighborhood is at least minpts, then q is de-fined as a core object. All objects within its eps-neighborhood are said to directly density reachable from q. In addition, an object p is said to density reachable it is with in the eps-neighborhood of an object that is directly density-reachable or density reachable from q. further more, objects p and q are said to be density connected if an object o exists that both p and q are density reachable [8]. These notions of density – reach ability and density connectivity are used to define what the DBSCAN algorithm considers as a cluster. A cluster is defined as the set of objects in a data set that are density – connected to a particular core object. Any object that is not a part of cluster is categorized as noise. This is in contrast to k- Means and Auto Class, which assign every object as a cluster.

B. *Explanation of Algorithm*

- DBSCAN requires two parameters: epsilon (eps) and minimum points (minPts). It starts with an arbitrary starting point that has not been visited. It then finds all the neighbor points within distance eps of the starting point.
- If the number of neighbors is greater than or equal to minPts, a cluster is formed. The starting point and its neighbors are added to this cluster and the starting point is marked as visited. The algorithm then repeats the evaluation process for all the neighbors recursively.
- If the number of neighbors is less than minPts, the point is marked as noise.
- If a cluster is fully expanded (all points within reach are visited) then the algorithm proceeds to iterate through the remaining unvisited points in the dataset.

C. *Working of DBSCAN Algorithm*

The DBSCAN algorithm works as follows. Initially, all objects in the data set are assumed to be unassigned. DBSCAN then chooses an arbitrary unassigned object p from the data set. If DBSCAN finds p is a core object, it finds all the density connected objects based on eps and minpts. It assigns all these objects to a new cluster. If DBSCAN finds p is not a core object , then p is considered to be noise and DBSCAN moves onto the next unassigned object. Once every object is assigned, the algorithm stops[8].

V. EXPRIMENTAL RESULT

Density-Based Spatial Clustering and Application with Noise (DBSCAN) was a clustering algorithm based on density. It did clustering through growing high density area, and it can find any shape ofTo perform the evaluation the following evaluation parameters are used. It is necessary to evaluate the performance of the system being designed. To do so the overall accuracy of the system is calculated. All most all the testing samples is determined. The overall accuracy is calculated and as shown in table 2.

We also determine the following factors.

- Over all Accuracy.
- Accuracy of each class.
- Precision values of each class.
- Recall values of each class.
- Fmeasure of each class.

We apply the following metrics to indicate the performance of our traffic classifier:

For a given class, the number of correctly classified objects is referred to as a True Positive (TP). The number of objects falsely identified as a class is referred as a False Positive (FP). The number of objects from a class that are falsely labelled as another class is referred as False Negative (FN).

Accuracy is the ratio of Sum of True Positive and True Negative to the sum of all True and False Positive, True and False Negative for all classes

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP + FP+TN+FN}}$$

Recall is the ratio of True Positives to the number of True Positives and False Negatives. This determines how many objects in a class are misclassified as something else.

$$\text{Recall} = \frac{\text{TP}}{\text{TP +EN}}$$

Precision is the ratio of True positives to True and False Positives. This determines how many identified objects were correct.

$$\text{Precision} = \frac{\text{TP}}{\text{TP +FP}}$$

F measures the balance between precision and recall; it is harmonic mean between them.

$$\text{F} = \frac{2\cdot\text{precision} \cdot \text{recall}}{\text{precision} \cdot \text{recall}}$$

Performance Evaluation

It is necessary to evaluate the performance of the technique being implemented. We evaluating the performance of classifier at number of clusters equal to 18. To do so Evaluation matrix is computed for test dataset and it is shown in Table 2. All the testing instances it is determined how many instances are incorrectly classified and correctly classified. It is difficult to determine the precise values of Eps and MinPts for DBSCAN [20]. However, we can use a probable range for their values by means of experience. The values of Eps are 1, 2, 4 and 8. The values of MinPts are 4, 6, 8 and 10.

Table 1: Confusion Matrix for Test Data set

	Predicted as				
Class	**Normal**	**Dos**	**U2R**	**R2L**	**Probe**
Normal	354	0	7	50	0
Dos	0	785	21	64	2
U2R	0	0	106	1	0
R2L	0	7	10	251	3
Probe	0	0	41	15	183

From Table 2 we calculate the overall accuracy of the classifier and it is 97.76% at number of cluster =18. Table 2 shows the accuracy, precision, recall and F measure value of each class. And fig. 2,3,4,5 are plotted from the Table 2. Several observations can be made from the table 2 First, more than 97% accuracy is achieved for all classes. Second, more than 94% precision achieved for all classes except U2R class. While normal class achieved 100% precisions i.e. the instances belong to other class are not classified as belongs to normal class. Third, the U2R class achieved lowest precision indicates that the other instances are misclassified as belongs to this class as compared to others. Forth, more than 76% recall achieved for all classes. Fifth, U2R class achieve 99.06% recall i.e. the instances belong to this class are more correctly classified. Sixth, Probe class achieved lowest recall values i.e. the large number of instances belongs to this class are misclassified as compared to the other classes. Seventh more than 72 % f-measure is achieved for all classes. Eighth R2L class achieved 91.26% f-measure value. This means that instances belong to this class are more correctly classified. Ninth U2R class achieved lowest f-measure values i.e. the large numbers of instances are misclassified as compared to other class.

Table 2: Precision, Recall and F_1 Measure

Class	Accuracy (%)	Precision (%)	Recall (%)	F_1 Measure (%)
Normal	98.57	100	86.13	92.54
Dos	97.65	99.11	90.02	94.35
U2R	98.00	57.29	99.06	72.60
R2L	96.25	94.76	99.15	96.90
Probe	98.47	97.34	76.56	85.71

VI CONCLUSION

In this paper we presented a DBSCAN based semi supervised clustering algorithm to build the classifier and it has been achieved successfully. It permits both labelled and unlabelled instances to be used in training the classifier. The classifier achieves 97.33% accuracy at K=18. It is observed that the accuracy of the classifier depends on the number of clusters and initial parameters i.e. minpts and epsilon distance of DBSCAN algorithm. This technique will be used in real time traffic classification in future.

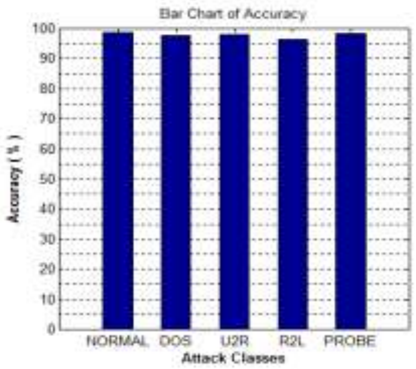

Figure 2: Accuracy of each class

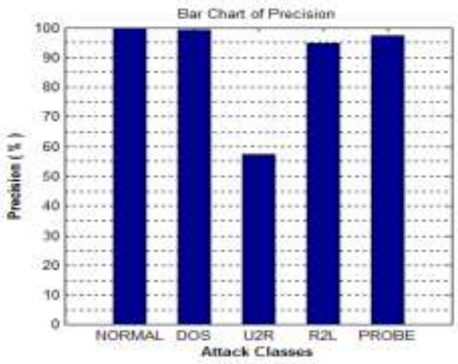

Figure 3: Precision of each class

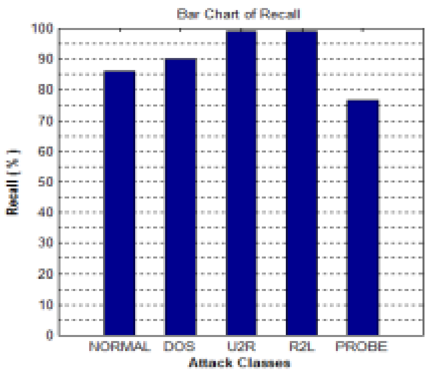

Figure 4: Recall of each class

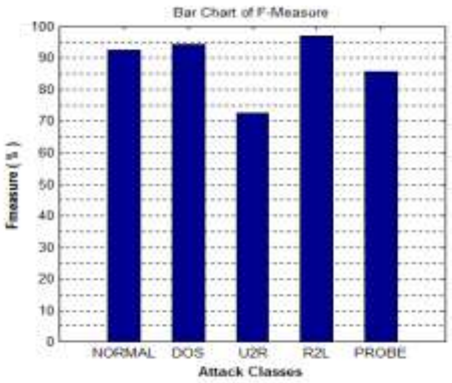

Figure 5: Fmeasure of each class

REFERENCES

[1] J. Erman, A. Mahanti, M. Arlitt, I. Cohen, and C. Williamson, "Semi-Supervised Network Traffic Classification",SIGMETRICS'07, June 12.16, 2007.

[2] J. Erman, A. Mahanti, M. Arlitt, I. Cohen, and C. Williamson, "Offline/Online Traffic Classification Using Semi-Supervised Learning", Technical report, University of Calgary, 2007.

[3] J. Erman, A. Mahanti, M. Arlitt, I. Cohen, and C. Williamson," Traffic Classification Using Clustering Algorithms", University of Calgary, SIGCOMM'06 Workshops September 1115, 2006, Pisa, Italy. Copyright 2006 ACM 1595934170/06/0009.

[4] Vinod Mahajan and Bhupendra Verma "Implementation of distance based semi-supervised clustering and probabilistic assignment technique for network traffic classification" March –April 2012.

[5] (Basic Book) H.Margaret , S. S. Dynham , "Data Mining Introductory and Advanced topics".

[6] NSL KDD Intrusion Detection Datasets. Available at: http://www.iscx.ca/NSL-KDD.

[7] "Design and Implementation of Semi-Supervised Classification using Support Vector Machine" GSITS M.E.Thesis , J. Thomas,2008.

[8] Caihong Yang, Fei Wang, Benxiong Huang " Internet Traffic Classification using DBSCAN" WASE Internetional conference of Information Technology 2009.

[9] Alberto Dainotti, Walter de Donato, Antonio Pescape and Pierluigi Salvo Rossi, "Classification of Network Traffic via Packet Level Hidden Markov Models", In IEEE GLOBECOM, New Orleans, LO, Dec.2008.

[10] IANA, "Internent Assigned Numbers Authority", http://www.iana.org/assignments/port-number.

[11] Subhabrata Sen, Oliver Spatscheck, and Dongmesi Wang, "Accurate, Scalable In- network Identificatio of P2P Traffic using application Signature", WWW 2004, New York, USA ACM, May 17-22, 2004, 512-521.

[12] I.Witten and E.Frank, Data mining: practical machine learning tools and techniques (Second Edition, Morgan Kaufmann Publishers, 2005).

[13] Thuy T.T. Nguyen, Grenville Armitage, " A Survey of Techniques for Internet Traffic Classification using Machine Learning", In IEEE communication surveys and tutorials, 2008, 1-21.

[14] Chuanliang Chen, Yunchao Gong and Yingjie Tian, " Semi-Supervised Learning Methods for Network Intrusion Detection", in proc. of IEEE International Conference on Systems Man and Cybernetics (SMC), 2008 , 2603-2608.

[15] Amita Shrivastav and Aruna Tiwari,"Network Traffic Classification using Semi-Supervised Approach", Second International Conference on Machine Learning and computing (ICMLC), 2010, 345-349.

[16] Rudra Pratap, Getting started with MATLAB 7 (OXFORD University press, Indian Edition).

[17] Nigel Williams Sebastian Zander, Grenville Armitage, Evaluating Machine Learning Algorihtms for Automated Network Application Identification, CAIA Technical Report 060410B, March 2006, 1-14.

[18] Levi Lelis and Jorg Sander,"Semi-Supervised Density- Based Clustering", in proc. of Nineth IEEE International Conference on Data Mining , Miami, FL , Dec 2009, 842-847.

[19] Liu Bin and Tu Hao, "An Application Traffic Classification Method Based on Semi-Supervised Clustering", A 2nd International Symposium on Information Engineering and Electronics Commerce (IEEC), 2010,1-4

[20] Caihong Yang, Fei Wang, Benxiong Huang "Internet traffic classification using DBSAN" A WASE International conference on Information Engineering, 2010