RESEARCH ARTICLE OPEN ACCESS

An Implementation Of Network Traffic Classification Technique Based On K-Medoids

Dheeraj Basant Shukla*, Gajendra Singh Chandel**
*(Department of Information Technology, S.S.S.I.S.T, Bhopal, India)
** (Department of Information Technology, S.S.S.I.S.T, Bhopal, India)

ABSTRACT
Classification of network traffic is extensively required mainly for many network management tasks such as flow prioritization, traffic shaping/policing, and diagnostic monitoring. Many approaches have been evolved for this purpose. The classical approach such as port number or payload analysis methods has their own limitations. For example, some applications uses dynamic port number and encryption techniques, making these techniques ineffective. To overcome these limitations machine learning approaches were proposed. But these approaches also have problems of labeled instances in supervised learning and tedious manual work in unsupervised learning. Our aim was to implement an approach for classification of network traffic on semi-supervised data which overcomes the shortcomings of other two approaches. In this approach, flow (instance) statistics are used to classify the traffic. These flow statistics contains few labeled and many unlabeled instances constitutes a training data set which was used for the training (learning) of classifier. Then we used two processes: the clustering (using K-Medoids) which divides the training data into different groups and classification in which the labeling to the groups was done. To build the model we used the MATLAB tool. To test the build model we used KDD CUP 99 intrusion detection data set, which includes both attack data and normal data.
Keyword **-** Classification, Clustering, Machine Learning, Semi-Supervised, K-Medoids.

I. INTRODUCTION

At present, the development of the TCP/IP technology based on Internet towards to a depth direction, such as the deployment of new generation infrastructure, the development of new technology; and the emergence of new application patterns and demands. Compared with the rapid development of the Internet, there is little research of network behaviors. Internet not only has the volatile, heterogeneity, dynamic, but also the strong society. The user behavior has an important effect on the Internet. So it is an interesting direction to understand such a system's statistical and dynamic property, and Internet users' behavioral character. In addition, research of Internet and users' behavior is an important step of many network management tasks [1].

Network Traffic Classification is the process of analyzing network traffic flows and classifies them mainly on the basis of protocols like TCP, UDP, IMAP, or POP3 etc or applications like games, messengers or news items etc. It is used to find out what types of application and protocols are run by end users [2,3]. Network traffic is a true reflection of the dynamics of network as it keeps track of all the activities of its user [1]. In this paper we will propose K-Medoids clustering algorithm for classification of network traffic using semi-supervised machine learning approach.

The remainder of the paper is organized as follows. Section II outlines Literature Review. Section III describes Proposed Methodology. Section IV gives details of Dataset and Analysis Tool. Section V gives Performance Evaluation and Analysis. Finally Section V concluded this paper.

II. LITERATURE REVIEW

The At least three historical developments over the last two decades have rendered less accurate the traditional method of using transport-layer (TCP and UDP) ports to infer most Internet applications (*port-based approach*)[5]:

- Top The proliferation of new applications that have no IANA registered ports, but instead use ports already registered (to other applications), randomly selected, or user-defined
- The incentive for application designers and users to use well-known ports (assigned to other applications) to disguise their traffic and circumvent filtering or firewalls
- The inevitability of IPv4 address exhaustion, motivating pervasive deployment of network and port address translation, where, for example, several physical servers may offer services through the same public IP address but on different ports.

At present, the main types of network application include HTTP, P2P, SMTP, POP3, Telnet, DNS, and

FTP, etc. This section discusses the level of traffic analysis, and demonstrates which levels we are concerned about. Meanwhile, several techniques presented in the literature are surveyed; such as Port Number Mapping, Payload-based Analysis and Machine Learning. Current research of network traffic analysis mainly focuses on the bit-level, packet-level, flow-level and stream-level. Bit-Level's research mostly concern network traffic's quantitative characteristics, such as network cable transmission rate and throughput's changes. Packet-Level cares the arrival procedure of the IP packet, delay and packet loss rate. Ref. [6] studied the change of the backbone network at flow load, round-trip time, packet disorder ratio and delay. The basis of flow partition is the address and protocol.

The traditional method relies on linking a well-known port number with a specific application, so as to identify different Internet traffic. The port-based method is successful because many well-known applications have specific port numbers (assigned by IANA[7]). For example, HTTP traffic uses port 80; FTP port 21.But with the emergence of P2P application, the accuracy of port-based is declined sharply. Because such application tries to hide from firewalls and network security tools by using dynamic port numbers, or masquerading as HTTP or FTP applications. So the port-based method is no longer reliable.

In order to deal with the disadvantages of the above method, a more reliable technique is to inspect the packet payload [8, 9]. In these methods, payloads are analyzed to determine whether or not they contain characteristic signatures of known applications. This technique can be extremely accurate when the payload is not encrypted. But this assumption is unrealistic because some P2P applications by use of different methods (encryption, variable-length padding), to avoid detecting by this technique. In addition, the demand of high process and storage capacity is discouraged, and privacy is concerned with examining user information [1].

Machine learning [10, 11], is one of the promising approach for traffic classification. There are two categories unsupervised and supervised in ML. The method in which the training data is labeled before is called as supervised learning. Labeled data means the input set for which the class to which it belong is known. The methodology in which the training data is unlabeled is called as unsupervised method. Unlabeled dataset is one for which class to which it belongs is unknown and is to be properly classified.

Another machine learning category is Semi-supervised methodology [12,13]. A learner and a classifier are two components of it. The learner is to distinguish a mapping between flows and traffic class from a training data set. Consequently, the classifier is obtained using this learned mapping. Fully labeled training data set is required to design the learner. It is very difficult as well as time consuming to obtain a fully labeled training data set. Quite the opposite, obtaining unlabeled training flows is reasonably priced. We build up and estimate a technique that allow us to design a traffic classifier using flow statistics using both labeled and unlabeled flows. Purposely, the learner is build using both labeled and unlabeled flows to show that unlabeled flow scan help to make the traffic classification problem handy. Semi-supervised approach is advantageous in the some situations. It is used to build fast and accurate classifier. This approach is vigorous and can lever formerly unseen flows. To improve the performance of the classifier it allows to add unlabeled flows. It classifies the given data set into appropriate classes using the k-means clustering algorithm [1, 4, 14, 15, 16].

III. PROPOSED METHODOLOGY

Our proposed system is based on clustering and probabilistic assignment technique using semi-supervised machine learning to analyze and classify network traffic on both labeled and unlabeled flow. Fig. 1 shows this proposed system, consisting of two major phases: Learning Phase and Classification Phase. In the learning phase, from training data set a mapping is determined between traffic flows and traffic class. Then, this mapping is used for classification in the next phase.

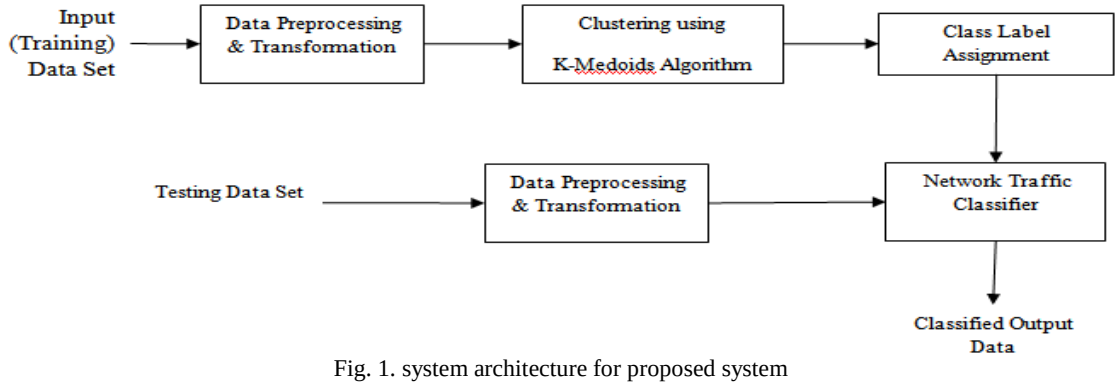

Fig. 1. system architecture for proposed system

3.1. Learning Phase

This phase of proposed system consist of mainly three subphases as shown in fig. 1.

- Preprocessing and Transformation
- Clustering using K-Medoids Algorithm
- Class Label Assignment

3.1.1. Preprocessing and Transformation

Normalization is used for data preprocessing, where the attribute values are scaled so as to fall within a small specified range such as 0.0 to 1.0. In this work for normalization the attribute values are divided by the largest value for that attribute present in the dataset [17].

3.1.2. Clustering

The *k*-means algorithm is sensitive to outliers because an object with an extremely large value may substantially distort the distribution of data. This effect is particularly exacerbated due to the use of the *square*-error function. Instead of taking the mean value of the objects in a cluster as a reference point, we can pick actual objects to represent the clusters, using one representative object per cluster. Each remaining object is clustered with the representative object to which it is the most similar. The partitioning method is then performed based on the principle of minimizing the sum of the dissimilarities between each object and its corresponding reference point. PAM (Partitioning Around Medoids) was one of the first *k*-medoids algorithms introduced. It attempts to determine *k* partitions for *n* objects. After an initial random selection of *k* representative objects, the algorithm repeatedly tries to make a better choice of cluster representatives. All of the possible pairs of objects are analyzed, where one object in each pair is considered a representative object and the other is not. The quality of the resulting clustering is calculated for each such combination. An object, ***oj***, is replaced with the object causing the greatest reduction in error. The set of best objects for each cluster in one iteration forms the representative objects for the next iteration. The final set of representative objects are the respective medoids of the clusters. The complexity of each iteration is $O(k(n-k)^2)$[18].

Algorithm: PAM (Partitioning Around Medoids), a K-Medoids algorithm for partitioning based on central object.

Input: k – Number of clusters

D – A data set containing n objects

Output: A set of k clusters

Procedure:

Step 1. Arbitrary choose k objects in D as the initial representative objects.

Step 2. Assign each remaining object to cluster with the nearest representative object.

Step 3. Randomly select a non-representative object, o_{random} Step 4. Compute the total cost, S, of swapping representative object, o_j, with o_{random}.

Step 5. If $S < 0$, then swap o_j with o_{random} to form the new set of k representative objects

Step 6. Repeat Step 2 to Step 5 until there is no change in medoid.

3.1.3. Class Label Assignment

Once training data is clustered available labeled flows i.e. clusters are mapped to different known classes. In this semi-supervised learning process, some clusters are mapped to different flow types. The collection of number of records are treated as input for classification process. The instance of a dataset is a record or a tuple (x,y), where attribute set is denoted by x and class attribute by y. A set of flows is suppose $X = \{X_1,\ldots\ldots..,X_N\}$. Xi is a flow instance, which is a vector of attribute values, $X_i = \{X_{ij} \mid 1 \leq j \leq m\}$, where m is the number of attributes, and X is the value of the j^{th} attribute of the i^{th} flow. The set of traffic classes are denoted by Y ,$Y = \{Y_1,\ldots\ldots\ldots.,Y_q\}$, where the number of classes are denoted by q. The Y i's can be classes. The mapping from m-dimensional variable X to Y forms a base for classification. The training is performed and the system is tested later on. The system is required to test on out of sample data. In the training phase center of the cluster is obtained. As well as in testing phase minimum distance of each record from centroid is compared, if found data is assigned to the same cluster.

By using clustering algorithm number of clusters are determined. A mapping from clusters to labels is done using probabilistic assignment technique. $P(Y = Y_j \mid C_k)$, where $j = 1,\ldots,q$. where q is number of class types and $k = 1,\ldots\ldots\ldots.,K$ where K is the number of clusters. The set of flows which are labeled to different applications of training data are used to find out probabilities, (xi; yi), $i = 1,\ldots\ldots,L$, where L = the total number of different labeled applications. $P(Y = y_j \mid C_k)$ is then estimated by the maximum likelihood estimate, njk nk , where njk is the number of flows that were assigned to cluster k with label j, and nk is the total number of (labeled) flows that were assigned to cluster k[4].

3.2. Classification Phase

In this phase, the traffic data obtained from first phase will be given to classifier for classification. In addition to evaluate classifier the testing dataset has all labeled flows which is preprocessed and transformed before it is given as input to classifier.

IV. DATASET AND ANALYSIS TOOL

4.1. KDD CUP 1999 Dataset

The KDD CUP 99 dataset is used for The Third International Knowledge Discovery and Data Mining Tools Competition, which was held in conjunction with KDD-99. Dataset contains 41 features and 1 class label [19].

The raw training data was about four gigabytes of compressed binary TCP dump data from seven weeks of network traffic. This was processed into about five million connection records. Similarly, the two weeks of test data yielded around two million connection records. A connection is a sequence of TCP packets starting and ending at some well-defined times, between which data flows to and from a source IP address to a target IP address under some well-defined protocol. Every connection is labeled as either normal or as an attack, with exactly one specific attack type. Attacks fall into four main categories [19]:

- DoS: denial-of-service, e.g. synchronous flooding;
- R2L: unauthorized access from a remote machine, e.g. guessing password;
- U2R: unauthorized access to local super user (root) privileges, e.g., various ``buffer overflow'' attacks;
- Probing: supervision and snooping, e.g., port scanning.

4.2. MATLAB

The name MATLAB stands for matrix laboratory. MATLAB is a high performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions ar expressed in familiar mathematical notation [19].

V. PERFORMANCE EVALUATION

The performance of classifier is greatly influenced by number of clusters. The number of clusters directly affects the quality of clustering, the time complexity. So, to evaluate performance of classifier the number of clusters has varied from 5 to 30. By this, we can determine at which number of cluster the classifier gives better performance.

We have conducted two experiments. In first experiment, we have chosen LD1 and CD1 as learning and classification dataset respectively. In second experiment, we have chosen LD2 and CD2 as learning and classification dataset respectively. On basis of above two experiments, we evaluate the f-measure.

To check effectiveness of classifier, we have calculated ratio of precision and recall represented with f-measure. The f-measure is defined as

$$\text{f-measure} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (1)$$

Table 1 shows that precision of classifier calculated for individual class at number of clusters equals to 30 for LD1 and LD2 (i.e. in Experiment 1 and 2).

Table 1. Precision of classifier at k=30

Attack Classes	Precision (%)	
	LD1	LD2
Normal	91.00	99.00
Probe	85.00	72.00
DoS	99.00	92.00
U2R	74.00	66.00
R2L	61.00	93.00

Table 2 shows that recall of classifier calculated for individual class at number of clusters equals to 30 for LD1 and LD2 (i.e. in Expriment 1 and 2).

Table 2. Recall of classifier at k=30

Attack Classes	Recall (%)	
	LD1	LD2
Normal	64.00	81.00
Probe	100.00	100.00
DoS	99.00	100.00
U2R	71.00	62.00
R2L	86.00	88.00

Table 3 shows that f-measure of classifier calculated from precision and recall.

Table 3. F-measure of classifier at k=30

Attack Classes	Recall (%)	
	LD1	LD2
Normal	75.14	89.10
Probe	91.89	83.72
DoS	99.00	95.83
U2R	72.46	63.93
R2L	71.37	90.43

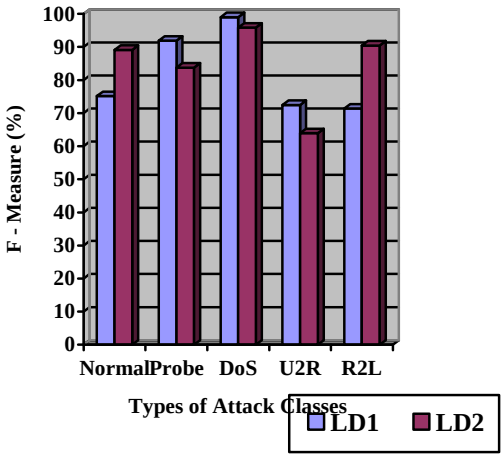

Fig. 2. Graphical Representation of F-Measure

The results shown in Table 3 and Fig. 2 indicates that the classifier correctly classifies the almost all instances belonging to class DoS and does not classify the other instances belonging to this class. Also there is no indication of new class detection.

VI. CONCLUSION

As an important character of network application, there is much literature about network traffic. In this work, from the micro perspective of network traffic, we described the use of semi-supervised machine-Learning to classify network traffic by application. The aim of the proposed system is to design and implement a semi supervised learning approach for network traffic classification based on clustering techniques. The system permits both labeled and unlabeled data to be used in training the network. The effectiveness of proposed approach has been evaluated by detailed experiments using different parameters. The experimental results has shown that the proposed technique fulfilled its said purpose and it is also possible to detect new classes during classification. It also significantly improves the accuracy and computational time. Experimental results shows that the performance of classifier can be improved by reducing the number of features in the dataset. In the future, this approach could become an excellent tool to classify network traffic.

REFERENCES

[1] Liu Yingqiu, Li Wei, Li Yunchun, "Network Traffic Classification using K-Means Clustering," in *Second International Multisymposium on Computer and Computational Sciences*, Aug.2007, pp. 360-365.

[2] Alberto Dainotti, Walter de Donato, Antonio Pescape, Pierluigi Salvo Rossi, "Classification of Network Traffic via Packet Level Hidden Markov Models," in *IEEE GLOBECOM*, New Orleans, LO, Dec. 2008, pp. 1-5.

[3] Geza Szabo, Istyan Szabo, Daniel Orincasy, "Accurate Traffic Classification," in *IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks (WoWMoM)*,Espoo, Finland, June 2007, pp. 1-8.

[4] Ms. Sheetal S. Shinde, Dr. Sandeep P. Abhang, "A Network Traffic Clasification Technique using Clustering on Semi-Supervised Data," in *International Journal of Electronics, Communication, & Soft Computing Sciences & Engineering*, ISSN: 2277-9477,Mar. 2012, pp. 151-155.

[5] Alberto Dainotti, Antonio Pescape, "Issues and Future Directions in Traffic Classification," in *IEEE Network*, Feb. 2012, pp. 35-40.

[6] C. Fraleigh, S. Moon, B. Lyles, C. Cotton, M. Khan, D. Moll, et al. "Packet-level traffic measurement from the sprint IP bacbone," in *IEEE Trans. On Networks*, 2003, 17(6):6-16..

[7] IANA, "Internet Assigned Numbers Authority," http://www.iana.org/assignment/port-numbers.

[8] C. Dews, A. Wichmann, A. Feldmann, "An analysis of internet chat systems," *In IMC'03*, New York: ACM Press, 2003:51-64

[9] P. Haffner, S. Sen, O. Spatscheck, D. Wang, "ACAS:Automated Construction of Application Signatures," *SIGCOMM'05 MineNet Workshop*, New York: ACM Press, 2005, PP. 197-202.

[10] Erman, A. Mahanti, and M. Arlitt. "Internet Traffic Identification using Machine Learning". in *Proc.GLOBECOM'06*, San Francisco,USA, November 2006.

[11] Sebastian Zander, Thuy Nguyen, "Automated Traffic Classification and Application Identification using Machine Learning", *Grenville Armitage Centre form Advanced Internet Architectures*, Swinburne, University of Technology ,Melbourne, Australia.

[12] J. Erman, A. Mahanti, M. Arlitt, I. Cohen, and C. Williamson, "Semi-Supervised Network Traffic Classification", *SIGMETRICS'07*, June 12.16, 2007, San Diego, California, USA. ACM 978-1-59593-639-4/07/0006.

[13] 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.

[14] Williamson," Traffic Classification Using Clustering Algorithms", University of Calgary, *SIGCOMM'06 Workshops*, September 1115, 2006, Pisa, Italy. Copyright 2006 ACM 1595934170/06/0009.

[15] Munz, Sa Li, G. Carle , "Traffic Anomaly Detection Using K-Means Clustering", *Computer Networks and Internet*, Wilhelm Schickard Institute for Computer Science,University of Tuebingen, Germany .

[16] Amita Shrivastav Aruna Tiwari , "Network Traffic Classification using Semi-Supervised Approach", *Second International Conference on Machine Learning and Computing*, © 2010 IEEEDOI 10.1109/ICMLC.2010.79.

[17] Vinod Mahajan, Bupendra Verma, ""Implementation of Distance Based Semi Supervised Clustering and Probabilistic Assignment Technique for Network Traffic Classification." *International Journal of Engineering Research and Applications*, no. 2 (2012): 1249-1252.

[18] Jiawei Han, Micheline Kamber, "*Data Mining : Concepts and Techniques*," Second Edition, MK Publishers,2006,pp 383-407.

[19] Vinod Mahajan, Bupendra Verma, "Implementation of network traffic classifier using semi supervised machine learning approach", *Nirma University International Conference on Engineering (NUiCONE)*, 2012, pp.1,6, 6-8 Dec. 2012