Data Traffic routing for Quality of Service

K.Raja
UG-Department of ECE
Kathir College of Engineering,
Covai, India - 636005.
dhivya3692@gmail.com

Prof. P.Kavitha.,M.E.,(Ph.D)
Department of Information Technology,
Kalaivani College of Technology,
Covai, India - 636005.

Abstract - Traffic classification is the major part to identify traffic based on the application in a large network. Here the traffic classification is useful to provide the Quality of Service (QOS), lawful interception and intrusion detection. The popular methods such as port and payload based techniques exhibit a number of limitations. Hence the research community uses the machine learning techniques. It analyzes the flow statistics to detect network applications. The statistical based approach is useful to assist in the traffic identification and classification process. The statistical features are flow size, flow duration, TCP port, packet inter-arrival times statistics, total number of packets, mean packet length, protocol, number of bytes transferred. There are two types of machine learning algorithms such as supervised and unsupervised algorithms. By using both the algorithm techniques, classify the traffic to identify when mixes up with other traffic. These two machine learning algorithms are analysed with datasets respectively based on the set of algorithms.

Keywords **– Machine Learning, Supervised, Unsupervised, Traffic Classification.**

I. INTRODUCTION

The Internet has become the backbone of human communication. Now a day's every electronic gadget is built to communicate over the Internet. Internet traffic is heterogeneous and consists of traffic flows from a variety of applications. In the current scenario online services such as email, social networks, multimedia communication, and https traffic have become an essential need for human beings. Many new applications emerge every day and are unique and have their own requirements with the respect to network criteria's. In an Internet user insight is also important. The user may not appreciate long waiting times, whereas the application can sustain large delays. This urges the research community to contribute in the classification of Internet traffic and treat the traffic fairly for meeting the user constraints at different level of abstraction in networking devices which includes routers, application gateway etc.

Real time traffic enforces delivery of real time traffic within a stipulated time period. Real time traffic flows are generated by applications like VoIP, Multimedia applications, Video Conferencing, Webinar, Online Gaming, IP-TV, Instant Messaging and interactive applications. It inflicts stringent demands to the Network. The most important task is the timely delivery of real time traffic to accumulate the original packets at the receivers' end. The efficiency of the network depends not only on bandwidth, packet loss, jitter and delay, but it also depends on the user satisfaction. But the performance of the real time traffic is greatly affected by the delay of individual packets. They are not able to adapt to a wide range of packet delay and delay variance at the transmission over data networks. Non real time traffic flows are generated by applications like E-mail, Peer to Peer etc. They are in sensitive to delay.Many Network operators want to manage their traffic such as real time traffic or business critical traffic which is given higher priority rather than non real time flows.

II. RELATED WORK

Identifying network flow using port numbers was traditional in the recent past. This approach was successful in the last decade because most of the applications use port numbers assigned by Internet Assigned Numbers Authority. This approach failed when the applications failed to communicate using their standard ports Karagiannis et al (2004). Applications that belong to the current generation use ephemeral ports or random ports and also use well known port numbers such as http, ftp etc to conceal them from Firewall or any tool that classifies the application.

Techniques that rely on inspection of packet contents choi et al(2004) have been proposed to address the diminished effectiveness of port-based classification. These approaches attempt to determine whether or not a flow contains a characteristic signature of a known

application. Studies show that these approaches work very well for today's Internet traffic, including P2P flows Haffner et al (2005).

Nevertheless, packet inspection approaches pose several limitations. First, these techniques only identify traffic for which signatures are available. Maintaining an up-to-date list of signatures is a daunting task. Recent work on automatic detection of application signatures partially addresses this concern Haffner et al(2005), Ma et al(2006) . Second, these techniques typically employ "deep" packet inspection because solutions such as capturing only a few payload bytes are insufficient or easily defeated. Deep packet inspection places significant processing and/or memory constraints on the bandwidth management tool. Packet inspection techniques fail if the application uses encryption. Many BitTorrent clients such as Azureus, µtorrent, and BitComet allow use of encryption.

The diminished effectiveness of the port-based and payload-based techniques motivates use of flow statistics for traffic classification Karagiannis et al (2004), Moore et al (2005). These classification techniques rely on the fact that different applications typically have distinct behaviour patterns when communicating on a network. For instance, a large file transfer using FTP would have a smaller inter arrival time between packets and larger average packet size than an instant messaging client sending short occasional messages to other clients can be distinguished from FTP data transfers because these P2P connections typically are persistent and send data bi directionally; FTP data transfer connections are non-persistent and send data only unidirectional. Although obfuscation of flow statistics is also possible, they are generally much harder to implement. There has been much work on scalable techniques for flow sampling and estimation Duffield et al(2002), Dufffieldet al(2004), Estan et al(2004), Kompella et al(2005), and furthermore, the logistics for collecting flow statistics is already available in many commercial routers NetFlow(2001) solution.

III. Machine Learning Techniques

The classification of flows is crucial in Network Management. It is used to improve the quality of service as well as network monitoring and control. There is a lot of contribution from the research community in classifying the flow type. Classifying the flow type is done using machine learning to build up a classifier to identify that traffic by packet statistics such as the maximum packet length, minimum packet length and standard deviation.

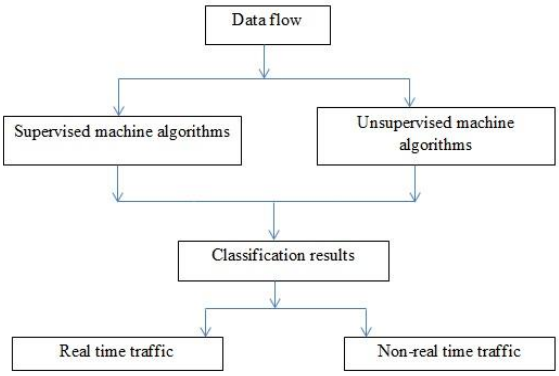

Figure 1. Classification Model

The machine learning techniques have two integral parts: 1. Supervised learning and 2. Unsupervised learning.

1. Supervised learning

With Supervised learning the class of traffic must be identified before it gets to be classified. The classification model that has been built using the training set of instances can able to predict the new instances by probing the feature values of unknown flows. The supervised learning uses weka to implement the algorithms. These algorithms' performance is calculated in terms of classification speed and the model building time.

A supervised learning algorithm analyzes the training data and produces an inferred function, which can be used for mapping new examples. An optimal scenario will allow for the algorithm to correctly determine the class labels for unseen instances. There are many algorithms. The following three are using in this project:

- Decision tree
- Naïve bayes tree
- Naïve bayes

A. Decision Tree

C4.5 Decision Tree algorithm creates a tree structured model where the nodes in the tree represent features and the branches represents values which connects features. A leaf node represents the class which terminates nodes and branches. The decision tree has been built with the root as a starting point and continuous down to its leaves. To classify the object, we begin at the root of the tree then compute the test and proceeds towards the branch which

yields a suitable outcome. This process prolongs until the leaf is met. If a class named by the leaf is identified then the object belongs to that class. The class instance can be determined by examining the path from nodes and branches to the terminating leaf. If all classes of an instance belong to the same class then the leaf node is labeled with that class. Otherwise the decision tree algorithm uses divide and conquer method which is used to divide the training instance set into non-trivial partitions until every leaf contain instances of only one class or until further partition is not possible. The tree will classify all instances if there is no conflicting. The prediction accuracy of unseen instances decreases by this over-fitting. To avoid this, some structures can be removed from the tree after it has formed. The decision tree algorithm is steadfast to classify and it is understandable because the data is split into nodes and branches. C4.5 is one of the most accurate classifiers and fastest classification speed.

B. Naïve Bayes

The Bayesian theorem is the basis of Naïve Bayes algorithm and this method is based on probabilistic knowledge. The Naïve Bayes classifier takes a sign from unrelated attributes to rustle up final prediction to classify the attributes. The Naïve Bayes classification uses Bayes rule to evaluate the conditional probability by examining the association between each attribute value and the class.

$$P(A/B) = \frac{\mathrm{P(B/A)P(A)}}{\mathrm{P(B)}}$$

Where A is class and B is fixed attribute value. To get the probability of an object which belongs to class A using these conditional probabilities multiplied together. Naive Bayes classifiers calculate the probabilities of a feature which is having a feature value. The frequency distribution cannot calculate the probability of a continuous feature if they have a large number of values. Instead it can be achieved by modeling features for the continuous probability distribution or discrete values. The classifier's performance was going down when it's considered full flow features. Instead, sub-flows are used to enhance the performance. Naïve Bayes uses two methods Naïve Bayes Kernel Estimation (NBKE) and Fast Correlation-Based Filter (FCBF) to minimize the future and improve the performance and time taken to build the classification model is less.

C. Naïve Bayes Tree

The Naïve Bayes Tree (NBT) is a combination of Decision Tree and Naïve Bayes classifier. The NBT algorithm is labelled as a decision tree which has nodes and branches and it is also defined as a Bayes classifier on the leaf nodes. The accuracy of both Naïve Bayes and decision tree are not good enough. The NBT algorithm is more accurate than C4.5 or Naïve Bayes on certain datasets. Like most other tree based classifiers NBT also has branches and nodes. The algorithm is mainly concerned with evaluating the utility of a split of each attribute. Utility for a particular node is calculated by making the data discrete and using the method called 5-fold cross validation for which Naïve Bayes is used to estimate the accuracy. The weighted sum of the utility of the nodes is considered as a utility of the split which is considered significant if there are at least 30 instances for the node and the relative error minimization is greater than five percent. The instances are divided based on the highest utility among all the attributes, if it is better than current node utility. If there is no such better utility a Naïve Bayes classifier is created for the current node. The NBT gains the advantage from decision trees and Naïve Bayes classifier because each node in the decision tree is built with univariate splits and Naïve Bayes is at the leaf node.

2. Unsupervised learning

The problem of unsupervised learning is that of trying to find hidden structure in unlabeled data. In order to classify the network traffic of unknown applications, it is a difficult problem to solve using supervised methods. There are many algorithms. The following three are using in this project:

- K-means
- Autoclass
- DBSCAN

A. K-means

K-Means algorithm is one of the unsupervised machine learning algorithms used to classify the traffic. It is a partition based clustering technique that tries to find out a user- specified number of clusters (K) which are denoted by using centroids. Euclidean distance is used to measure the similarity between flows. When the natural clusters are formed, the modeling step describes a rule to allocate a new flow to a cluster. The rule is that: The Euclidean distance is measured between new flow and the cluster. If the distance is minimal, then the new flow belongs to the cluster. The clusters are spherical in shape that is produced by K-Means algorithm. The training set contains the payload, so that the flows in each cluster are entitled with its source application. The learning output comprises of two sets: one set contains the explanation of

each cluster and the other contains the structure of its application. The online flows can be classified using these sets. In the classification phase, the first P packet sizes are captured and then compared it with the new flow. When the cluster is well-defined, the flow is related to the application that is more dominant in the cluster. The K-Means algorithm faces the challenges of categorizing the application when there is dominant of any of the clusters are not found. K-Means clustering algorithm is a simple and standard analysis method. The main objective is to divide n observations into K clusters, in which each observation fits to the cluster with the nearest mean. First select K initial centroids and each point is ascribed to the nearby centroids and each group of points is designated to the centroids is a cluster. Each cluster in the centroids is streamlined based on the points designated to the cluster. We repeat the update steps till the centroids keeps on same.

B. Autoclass

AutoClass is a probabilistic model based clustering technique. It automatically studies the natural cluster and soft clustering of data. Soft clusters slightly allocate the data objects to more than one cluster. To built the probabilistic model AutoClass uses Bayesian score to find out the best set of parameters that administer the probability distributions of each cluster. To achieve this, AutoClass uses EM algorithm and this algorithm is assured to meet the local maximum. In order to find out the global maximum the AutoClass execute repeated EM algorithm begins from pseudo random points in the parameter space. The parameter set which has the highest probability given the present database is selected as the best. AutoClass consumes time to build the model but it yields high accurate clusters.

C. **DBSCAN**

DBSCAN (Density Based Spatial Clustering of Applications with Noise) is a density based algorithms. It considers clusters as dense areas of objects that are detached by less dense areas. DBSCAN algorithm uses the views of density–reachability and density-connectivity. These clustering algorithms possess gain over partition based algorithms because they are not restricted to solve spherical shaped clusters but can able to solve random shapes. DBSCAN has the ability to find out the best set of clusters from the random shapes of cluster to minimize the amount of analysis required. The DBSCAN has two input parameters: epsilon (eps) which is the distance rounds an object that describes its eps-neighborhood. The given object is represented as q, when the number of objects located within eps-neighborhood is at least minPts, then q is called as core object. Objects within its eps-neighborhood named as directly density-reachable from q. Moreover, an object p is called as density reachable if it is within the eps-neighborhood that is directly density reachable or density reachable from q. additionally, p and q are called as density-connected and the same will be stated as density-reachable if an object o exits. These density-reachable and density-connectivity are used to explain the density based cluster algorithm. A cluster is a group of objects in a data set that are density connected to a specific core object. Any object that is not a portion of cluster is classifies as noise. DBSCAN produces lower accuracy and high precision.

IV. Summarised results

MACHINE LEARNING TECHNIQUES	TYPES	MERITS	ACCURACY	MODEL BUILDING TIME
SUPERVISED	DECISION TREE	steadfast to classify and it is understandable because the data is split into nodes and branches	More	less
	NAÏVE BAYES TREE	combination of decision tree and naïve bayes and it is useful to classify certain dataset which is not more accurately classified with decision tree or naïve bayes	More	more

	NAÏVE BAYES	**Based on probabilistic knowledge and takes a sign from unrelated to attributes to crack final prediction to classify the attributes**	**Less**	**less**
UNSUPERVISED	**K-MEANS**	**simple to classify and effectively classify large datasets**	**More**	**less**
	AUTOCLASS	**automatically determines the number of clusters**	**More**	**less**
	DBSCAN	**ability to detect noise and good efficiency on large database**	**Less**	**less**

V. Conclusion and Future Work

The dataset with flow statistical features is classified using machine learning algorithms such as supervised and unsupervised with set of algorithms respectively. By using both the machine learning techniques, identify the IP traffic and classify it when it is mixed up with other traffic. The supervised and unsupervised machine learning techniques classifies the dataset into real-time and non real-time traffic with set of algorithms respectively.The future work is to identify the algorithm which yields better results when comparing the results in terms of accuracy with one another in both the algorithm sets of supervised and unsupervised machine learning techniques. After identifying the better algorithm in both supervised and unsupervised algorithm sets, combine both the better algorithms together in order to yield more accurate results.

References

[1]. Erman.J, Arlitt.M and Mahanti.A, 'Traffic Classification using Clustering Algorithms', in proc of the SIGCOMM workshop on Mining network data, ACM, 2006.

[2]. Erman.J, Mahanti.A and Arlitt.M, 'Internet Traffic Identification Using Machine Learning', in proc. IEEE GLOBECOM, 2006.

[3]. Erman.J, Mahanti.A, Arlitt.M, Cohen.I and Williamson.C, 'Offline/Realtime Traffic Classification using Semi-Supervised Learning', Journal of Performance Evaluation, vol. 64, pp.1194-1213, 2007.

[4]. Gu.C, Zhuang.S, Sun.Y and Yan.J, 'Multi-levels Traffic Classification Technique', International Conference on Future Computer and Communication, vol.1, pp.448-452, 2010.

[5]. Jun Zhang,Yang Xiang, Yu Wang, Wanlei Zhou, Yong Xiang and Yong Guan, 'Network Traffic Classification Using Correlation Information', in IEEE Transactions On Parallel And Distributed Systems, vol. 24, no. 1, 2013.

[6]. McGregor.A, Hall.M, Lorier.P, Brunskill.J, 'Flow Clustering using Machine Learning Techniques', in proc of PAM, 2004.

[7]. Moore.A and Zuev.D, 'Internet Traffic Classification Using Bayesian Analysis Techniques', in SIGMETRICS, 2005.

[8]. Nguyen.T.T.T, Armitage.G, 'A Survey of Techniques for Internet Traffic Classification using Machine Learning', in IEEE Communications Surveys and Tutorials, 2008.

[9]. Risso.F, Baldi.M, Morandi.O, Baldini.A, Monclus.P, 'Lightweight, Payload-Based Traffic Classification: An Experimental Evaluation', 2008.

[10]. Sasan.A, 'Traffic Classification – Packet-, Flow- and Application-based Approaches', in International Journal of Advanced Computer Science and Applications, vol. 1, no. 1, 2010.

[11]. Sharma.R.K, Jain.A and Dubey.D, 'An Efficient ID3 Decision Tree for the Classification of Populated IP Address Using K-Mean Clustering', International Journal of Computer Science and Information Technologies, vol. 4, no. 1, pp. 159-162, 2013.

[12]. Yanai.R.B, Landberg.M, Peleg.D and Roditty.L, 'Realtime Classification for Encrypted Traffic',

in Proceedings of International Symposium SEA, pp. 373-385, 2010.

[13]. Zander.S, Nguyen.T and Armitrage.G, 'Automated Traffic Classification and Application Identification using Machine Learning', in LCN'05, Sydney, Australia, 2005.

[14]. Zander.S, Nguyen.T and Armitrage.G, 'Self-Learning IP Traffic Classification Based on Statistical Flow Characteristics', in PAM'05, Boston, USA, 2005.

[15]. Jun Zhang, Yang Xiang, Wanlei Zhou, Yu Wang, 'Unsupervised Traffic Classification Using Flow Statistical Properties And IP Packet Payload', in Journal of Computer and System Sciences, pp. 573-585, 2013.