INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH & DEVELOPMENT

ISSN 2278 – 0211 (Online)

A Network Traffic Classification Using C5.0 Algorithm

Om Singh
Computer Engineering Department
Dr. D. Y. Patil College of Engineering, Ambi, University of Pune, India
Prajakta A. Kumbhar
Computer Engineering Department
Dr. D. Y. Patil College of Engineering, Ambi, University of Pune, India
Kiran Shinde
Computer Engineering Department
Dr. D. Y. Patil College of Engineering, Ambi, University of Pune, India
Manzoor Shaikh
Computer Engineering Department
Dr. D. Y. Patil College of Engineering, Ambi, University of Pune, India

Abstract:
In this globalized era, the present scenario involves a high speed internet infrastructure. So the monitoring of the network traffic becomes a tedious task. Its analysis requires study and knowledge about various types of applications which form the network traffic. Various methods like Deep Packet Inspection (DPI), Machine Learning Algorithms (MLA's) have in practice over years for analyzing and classification of network traffic. This paper presents the concept of network traffic classification using C5.0 algorithm which is the latest algorithm out of the machine learning algorithms. It uses a C5.0 classifier and while capturing packets our machine learns itself by machine learning algorithm. A level of accuracy is obtained by using high quality of training data, a unique set of parameters are taken into account for both training and classification. This type of comparison is preferable over others as it is able to distinguish among seven different applications in a test set of range of seventy thousand to one lakh unknown cases with an average accuracy of 99.3-99.9%.

Key words: *Deep Packet Inspection (DPI), Traffic classification, Machine learning algorithms (MLA's), Computer networks, Boosting*

1. Introduction

The main concern of network monitoring is measuring the performance of the high speed network in a centralized way. The different types of networks carry data for many different kinds of applications which have their own requirements of performance. As soon as the system is connected to a network, the flow of packet starts right from there. Flows are generally considered to bidirectional i.e. one from the local machine or system to the server to which it is connected and other vice versa. A very simple approach to classify network is through realization of the group of packets having same IP address, same transport protocols and port number. This type of technique is very effective for protocols using fixed port number but in case of dynamic port number it doesn't work up to its expectations. On the other hand deep packet inspection (DPI) technique is quiet slow which requires a lot of processing power. Machine learning algorithms on the other hand require training data for their initial phase of learning. Our paper introduces C5.0 algorithm in traffic classification of the network and leaves a very clear impression about how it is better than other previous techniques and methodologies as it is able to give an accuracy of above 99% in the classification. The remaining part of this paper describes our work as we have tested two types of applications i.e. how packet is captured, trained and C5.0 is tested. The result is mainly displayed in the form of pie chart and bar chart.

1.1. Need for Network Traffic Classification

Classification of network traffic is the first way to identify various applications and protocols. Once the packets are classified as belonging to a particular application or protocol, they are marked or flagged. Marking is the process that colors the packets based on

certain classification policies, to provide appropriate treatment to those packets. It is needed to have a proper understanding of the applications and protocols in the network. It is also required in implementation of appropriate security policies. It helps in informing about the real attacks which is far below the false alarm rate. Last but not the least the prime concern lies in boosting the quality of service and analysis of the traffic.

1.2. Classification Attributes

Classification attributes generally fall under many criteria's. Generally using this type of algorithm for classification, attributes are divided into two types of disjoint sets. Actually flow of the packets is recognized on the basis of proportions of inbound to outbound payload bytes of the classified flow. Each flow is divided into X groups of Y packets where Y depends on the current iteration and X stands for the count of obtained groups. Each group from these two disjoint sets is used for the generation of one training case and another testing case for the classifier. Many features are taken into account like number of inbound/outbound/total payload bytes in the sample, ratio of all small inbound and outbound data packets, ratio of all large data packets, applications used etc. The other part exclusively contains protocol dependent attributes like which transport protocol is being used, number of ACK/PSH flags for inbound/outbound direction, local port and remote part etc.

2. C 5.0 Classifier

The C 5.0 algorithm is a descendent of C4.5 machine learning algorithm. Unlike C4.5 it is not memory wasting, time consuming and is easier to use. This algorithm is based on the decision trees. It is derived from an earlier system called ID3 where ID3 stands for Induction of Decision Trees. The decision trees are made out of list of possible attributes and set of training cases. These decision trees are used to classify thereof sets of test cases. The time used to generate the rules using this algorithm is much lower and it generates the rules which are even more accurate. The C5.0 classifier actually contains the simple command line interface which is used for the generation of decision tree rules and at the end finally testing of the classifier is done. The result is displayed mainly in the form of pie-chart and bar-chart. Many new techniques are introduced in C5.0, some of which are as follows-

- Boosting- It is a process of generation of several decision trees and they all are combined to improve the predictions.
- Marking of values can be done as missing or not applicable for particular cases.
- It supports cross validation.
- It supports sampling and makes it possible to avoid errors thus preventing any sort of harm or damage.

2.1. Project Methodology

Test challenges the assumptions, risks and uncertainty which inherit the work of the other disciplines, addressing those concerns by concrete demonstration and impartial evaluation. First, testing software is enormously difficult. The different ways a given program can behave are unquantifiable. Second, testing is typically done without a clear methodology so results vary from project to project, organization to organization: success is primarily a factor of the perseverance, skills, quality and talent of the individuals.

Third, insufficient use is made of productivity tools, making the laborious aspects of testing manageable: in addition to the lack of automated test execution, many test efforts are conducted without tools that allow the effect management of extensive Test Data and Test Results. While the extensibility of use and complexity of software makes complete testing an impossible goal, a well-conceived methodology and use of state of the art tools, can help to improve the productivity and effectiveness of the software testing.

2.1.1. Important Modules

- Start Capture packet:-This module enables the server to capture the packets from number of clients. In this module we capture the packet from high speed links which is done by the jNetPcap application. WinPcap is used for providing External java libraries which is used for collecting packets. WinPcap is used for support of windows environment. To capture the packet jNetPcap and WinPcap applications are used.
- Start Training**: -** This module is used to train to live captured packets, and those packets data are converted into byte format into text File after extracting packet features.
- Detection: - This module is used to detect the unknown packet in packet flow, and block that packet.
- Import dataset**: -** In this module, load the text File which is created in training part, and it displays the training data in tabular format.

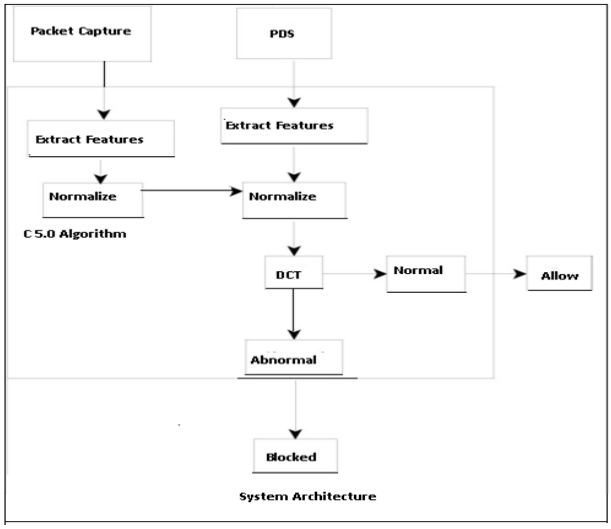


Figure 1

- Training c5.0:- In this module, machine learn itself according to training dataset which is created by extracting packet features. And it checks that whether that rule is present or not in training dataset. If it is present then it allows that packet otherwise block that packet.
- Test c5.0:- In this module, result will be displayed in the form of graphical format like Bar chart, pie chart etc.

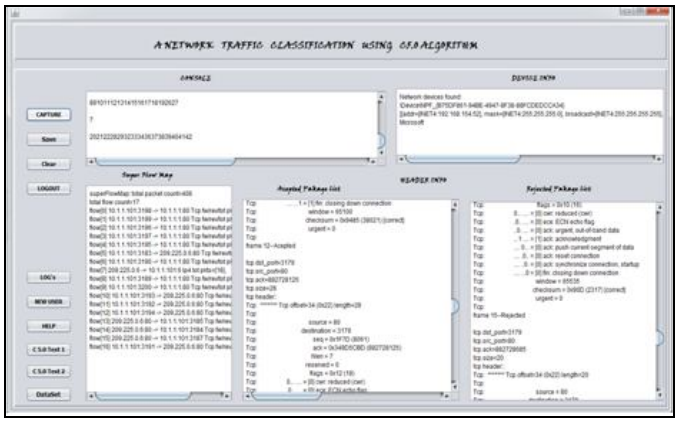
3. Comparison of Classification

Algorithm	Function	Description
Earlier methods (DPI, MLA, C4.5etc.)	classification	• Deep Packet Inspection (DPI) solution is quite slow and requires a lot of processing power. • It takes more memory space for storing data. • It requires more time classification of packets. • It requires large decision tree.
C 5.0 algorithm	classification	• C5.0 is faster than earlier methods like C4.5. • C5.0 is more memory efficient. • It supports boosting and gives them more accuracy. • 4. A C5.0 option automatically winnows the attributes to removes those that may be unhelpful.

Table 1

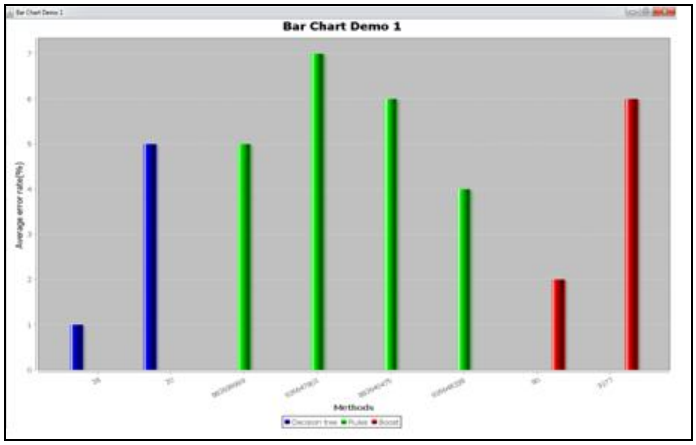
4. Result

The C5.0 classifier generates decision trees or classification rules based on the training cases provided. These rules are used to classify the test cases. We in our project have mainly focused on two protocols i.e. HTTP & TCP, thus mainly dealing with the network traffic coming through the web. Different attributes are used for classification and also different classification options are used. We have tested error rates of the provided classifiers and time needed to construct them. The lower error rate is achieved by using the boosted classifier. Our project shows that the classification error is higher when it involves many cases. We have derived our conclusion based on the statistics of 10 packets. First snapshot of our project deals with the capturing of the packets and give its associated properties like device information, transport protocol, number of packets, sequence and acknowledgement no., frame etc. Later on after training and testing the c5.0 classifier, the result is displayed in the form of bar chart, or pie chart. The snapshots are given in next page.

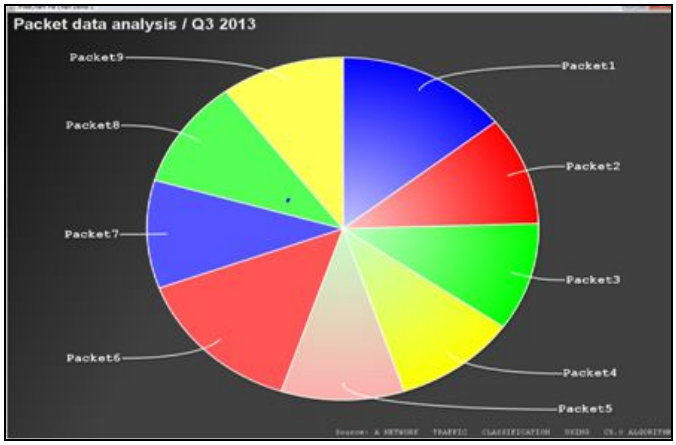


Snapshot 1: Network Traffic Classification Demo

Snapshot 2: Data Set For Training



Snapshot 3 :Average Error Rate Bar Chart



Snapshot 4: Packet Data Analysis Pie Chart

5. Conclusion

This project distinguishes different kind of traffic in computer network using C5.0 algorithm. Due to both training and testing the boosted classifier a high level of accuracy is obtained in the range of 93.5 to 99%. Since the result is generated in the form of pie-chat and bar-chart it is easily understandable.

Our main motivation for doing network traffic classification was to determine traffic demands, utilization and QoS of our network.

Our project is field for furthers improvements and in future we consider reducing or diminishing the misclassification among similar packets of FTP and Torrent File Transfer.

6. References

1. Tomasz Bujlow, KartheepanBalachandran, TahirRiaz, Jens MyrupPedersen,"Volunteer-Based System for classification of traffic in computer networks", Aalborg University
2. Tomasz Bujlow,TahirRiaz, Jens MyrupPedersen,"A Method for Assessing Quality of Service in Broadband Networks Section for Networking and Security", Department of Electronic Systems Aalborg University
3. TomaszBujlow,TahirRiaz, Jens MyrupPedersen,"Classification of HTTP traffic based on C5.0 Machine Learning Algorithm",Section for Networking and Security,Department of Electronic Systems Aalborg University
4. Jun Li, ShunyiZhang, Yanqing Lu, JunrongYan,"Real-time P2P TrafficIdentification", IEEE GLOBECOM 2008 PROCEEDINGS, pp. 1–5.
5. Jing Cai, Zhibin Zhang, Xinbo Song, "An Analysis of UDP TrafficClassification", 12th IEEE International Conference on CommunicationTechnology (ICCT), IEEE 2010, pp. 116–119.
6. Ying Zhang, Hongbo Wang, Shiduan Cheng, "A Method for Real-Time Peer-to-Peer Traffic Classification Based on C4.5", 12th IEEEInternational Conference on Communication Technology (ICCT), IEEE2010, pp. 1192–1195