Network Traffic Classification using Support Vector Machine and Artificial Neural Network

Ashis Pradhan
Department of Computer Science and Engineering, Sikkim Manipal Institute of Technology,
Majitar, Sikkim

ABSTRACT

The classification of Internet traffic has come to the forefront in recent times as organization of network traffic is necessitated by the increasing use of the internet and limited bandwidth. Also, network traffic classification finds its application in network security and for Qos (Quality of service). In this report, a certain number of flow features have been used as a basis for classifying the network traffic into various applications that run on the network as the classes. These flow features (13 in all) were extracted using a Perl script after capturing traffic using Wire shark. Seven network applications were chosen as the classes, visually, ftp, www, p2p, NetBIOS, dns, mail and telnet for classification. The machine algorithms that have been used for classification are Artificial Neural Network (ANN) and Support Vector Machine (SVM). These algorithms were used while designing a classification simulation model in WEKA in which Multilayer Perceptron (MLP) and sequential Minimal Optimization (SMO) function was used respectively. Furthermore, comparisons on the performance of these algorithms have been carried out for arriving at the better network traffic classification..

Keywords

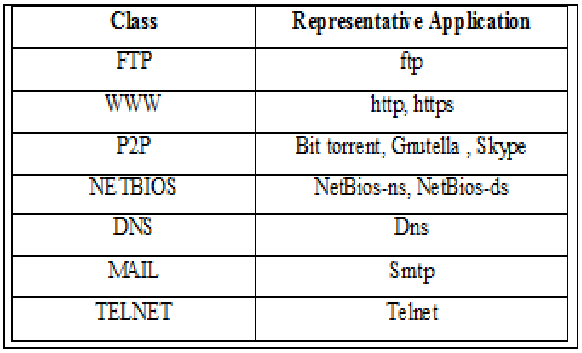
Support Vector Machine, Artificial Neural Network, Machine learning algorithm, Quality of Service, Security perspective, Network Traffic Classification.

1. INTRODUCTION

Internet traffic and control have attracted an increased amount of interest in the past few years. From the security perspective, the fast identification of malicious traffic can help for security control and isolation of attackers. From the quality of service (QoS) perspective, the accurate classification of different traffic can help to identify the application utilizing network resources and facilitates the instrumentation of the quality of service (QoS) for different applications. Furthermore, network operator can trace the growth of different application and provide network accordingly to accommodate the various needs of the population. This report basically discusses the classification of network traffic into broad categories of applications for improving the quality of service using machine learning algorithms and comparing their performance.

From the resource utilization and quality of service (QoS) requirement, the network applications were divided into few categories based on the data captured. The Table 1 shown below shows the different application based on which we have categorized different classes for classification process.

Table 1: Internet Traffic Classes



Class	Representative Application
FTP	ftp
WWW	http, https
P2P	Bit torrent, Gnutella , Skype
NETBIOS	NetBios-ns, NetBios-ds
DNS	Dns
MAIL	Smtp
TELNET	Telnet

Basically, this report deals with the supervised type of machine learning technique for the classification of network traffic. The classification of the network traffic into different class is based on application flowing in the network. The pre-processing of dataset is very essential part for any supervised type classification of network traffic. Initially, a dataset was prepared manually by collecting the raw network data in promiscuous mode using tool Wireshark and pre-processing it using Perl script to extract specific features from the packet header. Then this dataset is used for classification purpose in the simulation model that was designed in WEKA. The preparation of training dataset is most crucial part as it is very much necessary to initialize each different class manually based on the flow-wise properties, which is a very time consuming process.

The two most important machine learning algorithms, namely, SVM and ANN, were used for classifying the network traffic into above stated different classes. The SVM and ANN were chosen, as this algorithm has proven to be most efficient machine learning algorithms through recent studies and is used mostly for classification purpose. Finally, later in this report, a comparative study of the classification result of SVM and ANN is given after implementing prepared dataset on the simulation model that was designed using WEKA3.6.1. This classification simulation model includes two main modules i.e., SMO and MLP that implements SVM and ANN algorithm respectively. Also, this report gives a brief idea of the performance given by the implemented algorithm for better classification and also discussed some of the related future works.

The Figure 1 shown below is the methodology that has been followed for accomplishing the classification of network traffic in different classes.

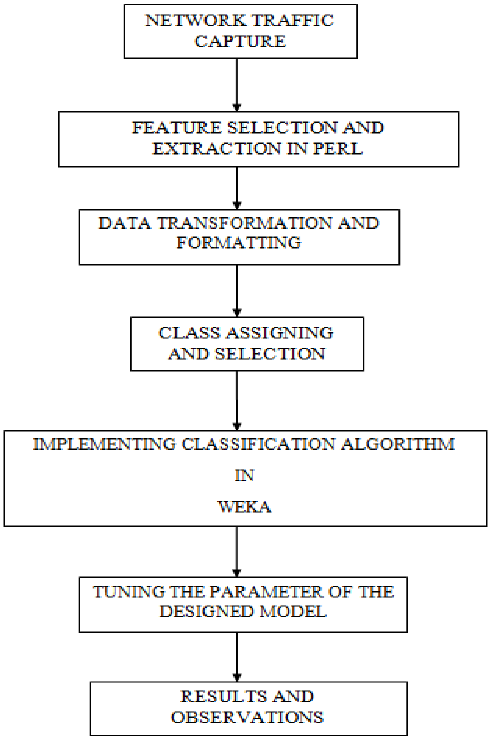

NETWORK TRAFFIC CAPTURE
FEATURE SELECTION AND EXTRACTION IN PERL
DATA TRANSFORMATION AND FORMATTING
CLASS ASSIGNING AND SELECTION
IMPLEMENTING CLASSIFICATION ALGORITHM IN WEKA
TUNING THE PARAMETER OF THE DESIGNED MODEL
RESULTS AND OBSERVATIONS

Figure 1: Methodology Flow Diagram

2. RELATED RESEARCH

The researchers [1], have talked about cyber-attack that has predominantly been reactive. They mostly focus on monitoring network traffic, detecting anomalies and cyber-attack traffic patterns, combating cyber-attacks and mitigating their effects. They presented their work on machine learning-based classification technique to identify command and control traffic of IRC-based botnets with a compromised set of hosts that are collectively commanded using IRC (Internet Relay Chat). They divided their work into (a) distinguishing between IRC and non-IRC traffic and (b) distinguishing between botnet and real IRC traffic.

Application of Machine learning for network traffic classification [2] has been dealt, in which the researchers have created an extensive E-dare system for early detection of malicious code in network traffic.

The work [3] gave the idea of feature selection, in which Traffic classification has been critically examined by conducting a thorough evaluation of three classification approaches, based on transport layer ports, host behaviour, and flow features. In this work, the researchers have mostly talked about different flow features that they have taken in their experiment. This work deals with the broad range of data against which they tested the three classification approaches, visually SVM, BLINC and a port based classification technique. They also have analyzed the advantages and limitations of each approach, evaluated methods to overcome the limitations, and extracted insights and recommendations for both the study and practical application of traffic classification.

The classification the network traffic into several broad classes like WWW, mail, attack, P2P, etc, were conducted by Andrew W. Moore[4]. He classified the network traffic into broad categories of applications based on the similar properties of same class. A naive Bayes classification algorithm is used to classify the network traffic in which he used the Bayes formula is used to calculate the posterior probability of a testing sample and select the largest probability class as the classification result. His work gave the brief outline of the classification of network traffic.

The Ruixi Yuan, Zhu Li & Xiaohong classified the network traffic into broad categories of application using SVM based machine learning method. They suggested that the accurate and timely traffic classification is critical in network security monitoring and traffic engineering. They also told that the traditional methods based on port numbers and protocols for classification have proven to be ineffective in terms of dynamic port allocation and packet encapsulation. Also, the signature matching methods could only handle the signatures of a limited number of IP packets in real-time. They have basically dealt with the methods that classify the Internet traffic into broad application categories according to the network flow parameters obtained from the packet headers. An optimized feature set is obtained via multiple classifier selection methods. Their experimental results using traffic from campus backbone showed that an accuracy of 99.42% was achieved with the regular biased training and testing samples, and an accuracy of 97.17% was achieved when un-biased training and testing samples were used with the same feature set. Finally, they suggested that since all the feature parameters are computable from the packet headers, the proposed method is also applicable to encrypted network traffic.

The researchers [7] studied on the network security risk evaluation and analysed the traditional risk evaluation methods, then they proposed a new network security risk evaluation method based on Support Vector Machine (SVM) and Binary tree. Unlike the traditional risk evaluation methods, SVM proved to be a novel type of learning machine technique which is developed on structural risk minimization principle. SVM has many advantages in solving small sample size, nonlinear and high dimensional pattern recognition problem. The principles of SVM and binary tree are introduced in detail and applied it into network security risk assessment. Compared to ANN about the Classification precision, Generalization Performance, learning and testing time, they proved that the SVM method has higher Classification precision, better generalization Performance and less learning and testing time, especially get a better assessment performance under small samples. Their results indicate that SVM has absolute superiority on network security risk evaluation, the validity and superiority of this method is approved through the experiment.

The book Neural Network [8] gave the basic idea to understand the basics of algorithm and also the Herv´e Abdi [9] threw some light on the same. The Vikramaditya Jakkula [10] gave overall concept of SVM with respect to classification, which describes only the theoretical aspect for pattern classification

Another introduction to Neural Networks and the few of its basic technicalities have been given and discussed in a teaching package by Carlos Gershenson [11]. In this work, he tried to clarify the working of the back propagation algorithm and suggested that the back propagation algorithm is used by layered feed forward Neural Networks.

The researcher, Ian H. Witten, discussed the classification learning. In Classification learning, he presented an algorithm with a set of classified examples or ''instances'' from which it was expected to infer a way of classifying unseen instances into one of several ''classes''. He told that Instances have a set of features or ''attributes'' whose values define that particular instance, and the numeric prediction, or ''regression,'' is a variant of classification learning in which the class attribute is numeric rather than categorical. Finally, he said that the Classification learning is sometimes called supervised because the method operates under supervision by being provided with the actual outcome for each of the training instances.

The Chintan Trivedi [13] used a Neural Network to classify Internet traffic based on the application to which the traffic packets belong. The scheme that he followed for classification is the use of statistical information which does not involve reading of any packet headers for determining the application. The classification is done with the Artificial Neural Network using a conventional feed-forward back propagation network with three layers. The classification results given by he, indicates that artificial neural networks exhibit a strong potential for use in applications involving classification of Internet traffic flows.

The researchers also have worked with SVM and ANN method both, where they analyzed their each individual performance. The Nello Cristianini and John Shawe-Taylor [14], gave the comparative study on the result of SVM and Neural Network for network security risk evaluation where they found that the SVM showing the better performance than Neural Network when the size of the pattern increases drastically.

3. FEATURE EXTRACTION

One of the most crucial steps in network traffic classification is the feature extraction process. It is also known as pre-processing steps in network traffic classification. In order to complete this task successfully a different language called PERL is used that can handle the network packet carefully. Perl is an interpreted high-level programming language developed by Larry Wall.

The total number of features extracted from Perl script is 13 which include flow, total number of packet per flow, arrival time, packet length, protocol, source and destination ip, source and destination port number, Ethernet-type, inter-arrival time, total packet length, average packet length, average inter-arrival time and variance for inter-arrival time. Initially, the program designed was able to extract features in packet-wise but later it was changed to extract features in flow-wise order to determine the number of flows flowing during one session and thus making our calculation easier.

In order to extract the features mention above, it is necessary to decode the raw packet that was captured and the syntax for decoding the packet is NetPacket::Ethernet->decode ($packet).

The syntax for extracting features from packet header for some of the above mentioned features is shown below:

- For arrival time, *$header-> {tv_usec}.*
- For packet length, *$header-> {len}.*
- For source ip, *$ip_obj-> {src_ip}.*
- For destination ip, *$ip_obj-> {dest_ip}.*
- For protocol used, *$ip_obj-> {proto}.*
- Source port number, *$frame-> {src_port}.*
- Destination port number, *$frame-> {dest_port}.*

In order to extract features like ip, protocol and port number, it is necessary to decode the data part of the packet and is done using "NetPacket::IP->decode ($eth_obj-> {data})". The "$ip_obj" is an object variable used for extracting ip, port and protocol information. Lastly, the port number is extracted based on the protocol used i.e. (TCP, UDP or ICMP).The remaining features like inter-arrival time, total packet length, average packet length, average inter-arrival time and variance are calculated manually using formula given below, for analysing variations in different flows:

- Inter-arrival time $_i$ = (Arrival Time $_i$ - Arrival Time $_{i-1}$) + Inter-arrival time $_{i-1.}$
- Avg. Inter-arrival time $_i$ = Inter-arrival time $_i$ / total number of packets (flow$_i$).
- Variance (Inter-arrival time) = E(x^2) - $(E(x))^2$.
- Total packet length $_i$ = (current packet length) $_i$ + (Total packet length) $_{i-1}$.
- Average packet length $_i$ = Total packet length $_{i\,/}$ total number of packet (flow$_i$).

Where, i = index for the current flow.

After extracting all the features mentioned above, the Perl script saves the output in simple ".txt" file. But such type of file format cannot be accepted by the designed model. So, next a dataset was prepared in such a format that can be accepted by the designed simulation model. Basically, the model is designed to accept accepts two types of input data format namely, ".arff" and ".csv". As for convenience to store data, the ".csv" format is used to store flow features as an input to the simulation model. To do such, an ms-excel sheet was used to import data from the "txt" file to excel sheet and saving it in comma delimited "csv" format.

4. EXPERIMENT DETAILS

After preparing a proper dataset, the classification simulation model was built in WEKA3.6.1. The designed model includes both MLP and SMO for implementing ANN and SVM respectively, in a single simulation model itself for studying the classification results comparatively and hence, analyzing it. The Knowledge flow Interface from WEKA was used to build a model as it is more flexible and easy to work with it and in which the flow of data can be known easily between different modules.

The important modules used while designing of model are:

- The CSV Loader: It loads the data stored in a '.csv' formatted input file.
- The Class Assigner: It assigns the nominal value based on which classification is done. By default, the last column of dataset is taken for nominal value.
- Train Test Split-maker: Divides the dataset file to separate training set and test set. The training set is used for training while the test set is used to evaluate the performance of the classifier in question.
- Classifier Performance Evaluator: As the name suggests, it evaluates the performance of the classifier used.
- Text Viewer: Conveys the output of the Classifier Performance Evaluator in a text form.
- Model Performance Chart: Conveys the output of the Classifier performance evaluator as a graphical representation.

In addition to these modules, there is another module which represents the classifier. This classifier module is adjusted between the Train Test Split-maker and the Classifier Performance Evaluator. For the purpose of classification of network traffic, the classifiers used were MLP (for ANN), SMO (for SVM). The following Figure 2 shown below is the knowledge flow layout or the simulation model designed for the purpose of classifying network traffic for a given dataset.

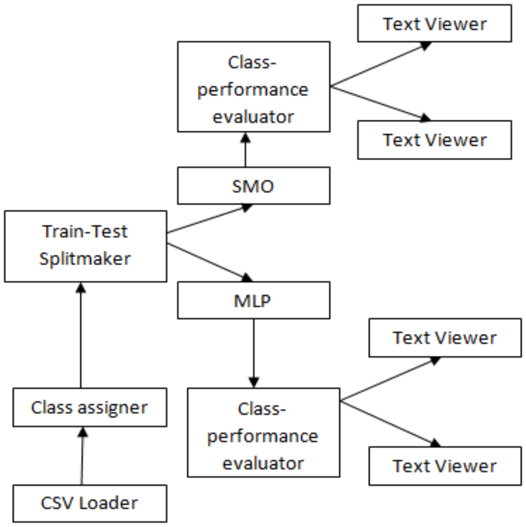


Figure 2: SVM and ANN Model

Here, exactly the same dataset was used for classification which is further divided into training set and test set and thus providing these dataset to each classifier for training and testing purpose. The results are interpreted mainly from the output of the Text Viewer of each individual algorithm.

5. RESULTS AND DISCUSSIONS

Basically, the results that were obtained are in two form, text view and graphical view. The text view shows the details of the results in textual form and also gives the confusion matrix to determine how correctly the algorithm classifies. The graphical view shows the model performance chart in graphical representation.

In order to get the proper result, it is very much necessary to supply proper input data in a correct format to the model developed so far. The input data that was supplied to simulation model is in ".CSV" format.

The comparison between ANN and SVM algorithms was used in WEKA, visually the MLP and SMO algorithms. Firstly, the standard results of individual algorithms were compiled, where the dataset used had 454 instances and the train percent was taken to be 66%. The configuration of both MLP and SMO modules were changed to enhance the results. The results for classification of SVM and ANN are shown below in Table 2.

Table 2: Comparison between ANN and SVM.

	Standard Dataset = 454 Instances **Train Percent = 66.0**	**Reduced Dataset = 47 Instances** **Train Percent = 66.0**	**Standard Dataset = 454 Instances** **Increased Train Percent= 80.0**
ANN	85.443	82.3529	86.8132
SVM	91.1392	88.2353	87.9121

The first result is the standard results with default value in classifier and in train-test split maker. The next is the result when the size of the dataset is decreased from 454 instances to 47 instances, but keeping the same value of train-test split maker. Finally, the train percent was increased and then observe the difference in the results obtained.

From the above result, it was seen that SVM algorithm always exhibits a better performance than the ANN algorithm. Upon reducing the dataset, we can clearly observe the decrease in performance of both the algorithms. However, on increasing the train percent, we can see how the ANN algorithm improves itself drastically in contrast to the SVM algorithm. This shows that the ANN algorithm is much more dependent on the training data as compared to the SVM algorithm.

The performance accuracy for these two algorithms is shown in Figure 3 below:

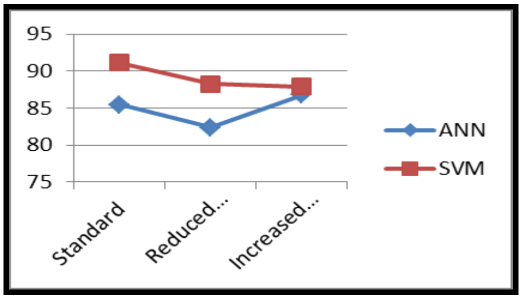


Figure 3: Performance graph for ANN & SVM

6. CONCLUSION

To summarize and draw down the conclusion, the Network Traffic flows were classified successfully into the various applications that they belong to, using the machine learning techniques of SVM and ANN. A comparative study on the performance of the different algorithms has been done and then analyzed. In the whole process, tools like Wireshark (to capture network packets) were used, a Perl script (to extract the features set and distributing the packets into various flows), data preprocessing tools like Microsoft Word Excel (to prepare the dataset for classification), and finally WEKA3.6.1 (where the classification model was built using the MLP and SMO modules) were used.

The limitation of this work is that only 7 applications have been taken into consideration as the classes for classification and only 13 fundamental flow features were used for classification purpose. The classification is done in WEKA3.6.1, which is only a simulation tool.

As an improvement for future, P2P packets from various applications can be captured for the purpose of P2P classification. P2P classification may be applied to traffic regulation as recent P2P applications tend to acquire a greater portion of Network Traffic and hence decreasing network throughput. Furthermore, upon acquiring the necessary data, malicious network traffic like those of Botnet, can also be detected in a network using the same techniques.

7. REFERENCES

[1] Carl Livadas, Robert Walsh, David Lapsley, W. Timothy Strayer, "Using Machine Learning Techniques to Identify Botnet Traffic".Proceedings of the Second IEEE LCN Workshop on Network Security (WNS), November 14, 2006, Tampa, Florida.

[2] Yuval Elovici, Asaf Shabtai, Robert Moskovitch, Gil Tahan, and Chanan Glezer, "Applying Machine Learning Techniques for Detection of Malicious Code in Network Traffic". Proceedings of the 30th annual German conference on Advances in Artificial Intelligence, Publisher: Springer, Volume: 4667/2007, DOI: 0.1007/978-3-540-74565-5_5, pp. 44-50, 2007.

[3] Hyunchul Kim, K. C. Claffy, and Marina Fomenkov "Internet Traffic Classification Demystified: Myths, Caveats, and the Best Practices". Proceedings of the 2008 ACM CoNEXT conference, Publisher: ACM, ISBN: 9781605582108, DOI: 10.1145/1544012.1544023,pp.1–12, 2008.

[4] Andrew W. Moore and Denis Zuevy "Internet Traffic Classification Using Bayesian Analysis Techniques". ACM SIGMETRICS Performance Evaluation Review, Volume: 33, Issue: 1, Publisher: ACM New York, NY, USA, ISBN: 1-59593-022-1, pp. 50-60, 2005.

[5] Ruixi Yuan & Zhu Li & Xiaohong Guan & Li Xu "An SVM-based machine learning method for accurate internet traffic classification". Information Systems Frontier, Springer Science + Business Media, Volume 12, Number 2, pp. 149-156, DOI: 10.1007/s10796-008-9131-2, 2008.

[6] Zhu Li, Ruixi Yu and Xiaohong Guan "Accurate Classification of the Internet Traffic Based on the SVM Method". Proceedings of IEEE International Conference on Communications, ICC 2007, Glasgow, Scotland, 24-28 June 2007. IEEE 2007.

[7] Li Cong-cong, Guo Ai-ling and Li Dan "Application Research of Support Vector Machine in network Security risk Evaluation". Intelligent Information Technology Application Workshops (IITAW), Issue Date: 21-22 December 2008, ISBN: 978-0-7695-3505-0, pp. 40- 43, 2008.

[8] Simon Haykin, "Neural Networks – A Comprehensive Foundation". McMaster University - Hamilton, Ontario, Canada, 2nd Edition.

[9] Herv´e Abdi, "Neural Networks". The University of Texas, Dallas.

[10] Vikramaditya Jakkula, "Tutorial on Support Vector Machine (SVM)".School of EECS, Washington State University, Pullman 99164.

[11] Carlos Gershenson, "Artificial Neural Networks for Beginners". Cited as: arXiv:cs/0308031v1, 20 Aug 2003.

[12] Ian H. Witten, "Classification". Department of Computer Science, University of Waikato, Hamilton, New Zealand, August 2009.

[13] Chintan Trivedi, Mo-Yuen Chow Arne Nilsson H. Joel Trussell, "Classification of Internet Traffic using Artificial Neural Networks". Publisher: North Carolina State University, Center for Advanced Computing and Communication, series/report no: TR-02/05, year: 2002.

[14] Nello Cristianini and John Shawe-Taylor, "An Introduction to Support Vector Machines and Other Kernel-based Learning Methods". Publisher: Cambridge University Press New York, NY, USA, ISBN: 0-521-78019-5, pp. 189, year: 2000.

[15] Tom Mitchell, "Machine Learning". Publisher: McGraw-Hill Computer science series, 1st Edition, ISBN: 0070428077, March 1997.

[16] V.K. Pachghare, Dr. Parag Kulkarni, "Research on computer network security based on Pattern Recognition". IJCSNS, International Journal of Computer Science and Network Security, VOL.8 No.10, October 2008.